DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

RE-ENGINEERING A COMPUTER-BASED TRAINER FOR A HELICOPTER NIGHT VISION SYSTEM

by

Dabney R. Kern
Kenan J. Shaffer

September, 1996

Co-Advisors: Anthony Ciavarelli
Kishore Sengupta

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1996	Master's Thesis

4. TITLE AND SUBTITLE
RE-ENGINEERING A COMPUTER-BASED TRAINER FOR A HELICOPTER NIGHT VISION SYSTEM

5. FUNDING NUMBERS

6. AUTHOR(S) Kern, Dabney R. and Shaffer, Kenan J.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
NAVAIRSYSCOM, PMA-205; 1421 Jefferson Hwy; Arlington, VA 22243

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(maximum 200 words)***

While there have been unfavorable responses to Department of Defense "right-sizing" efforts, this force restructuring has actually produced certain positive results. Capitalizing on technological advances, the aviation community, in particular, has adapted to personnel cuts and reduced budgets without sacrificing the quality of training. As a result, considerable emphasis is currently placed on computer-based training (CBT) applications. The development of this type of training for critical, high-risk, missions, such as those involving scarce night vision equipment, has encouraged numerous research projects including this thesis.

Sponsored by Naval Air Systems Command (PMA-205), this thesis discusses methods used to re-engineer the UH-1N helicopter Aviator's Night Vision Imaging System/Heads-Up Display (ANVIS/HUD) CBT for use in the HH-60H community. By using portions of code, graphics, and text originally designed for the UH-1N CBT, the HH-60H version was developed through a revision process which incorporated new material as required.

The final product is a trainer consisting of five instructional modules, combining student evaluation and remediation features through interactive lessons and exercises. In accordance with current design principles, an object-oriented authoring system enabled the production of a quality CBT that meets the sponsor's budget and time constraints, and promises to be a key training asset for HH-60H personnel.

14. SUBJECT TERMS Computer-Based Training, Night Vision Goggles, Re-engineering			15. NUMBER OF PAGES 98
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

Approved for public release; distribution is unlimited.

RE-ENGINEERING A COMPUTER-BASED TRAINER FOR A HELICOPTER NIGHT VISION SYSTEM

Kenan J. Shaffer
Lieutenant Commander, United States Navy
B.E.E., Villanova University, 1984

Dabney R. Kern
Lieutenant, United States Navy
B.S., Jacksonville University, 1989

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
September 1996

Thesis
S4335
c.1

ABSTRACT

While there have been unfavorable responses to Department of Defense "right-sizing" efforts, this force restructuring has actually produced certain positive results. Capitalizing on technological advances, the aviation community, in particular, has adapted to personnel cuts and reduced budgets without sacrificing the quality of training. As a result, considerable emphasis is currently placed on computer-based training (CBT) applications. The development of this type of training for critical, high-risk, missions, such as those involving scarce night vision equipment, has encouraged numerous research projects including this thesis.

Sponsored by Naval Air Systems Command (PMA-205), this thesis discusses methods used to re-engineer the UH-1N helicopter Aviator's Night Vision Imaging System/Heads-Up Display (ANVIS/HUD) CBT for use in the HH-60H community. By using portions of code, graphics, and text originally designed for the UH-1N CBT, the HH-60H version was developed through a revision process which incorporated new material as required.

The final product is a trainer consisting of five instructional modules, combining student evaluation and remediation features through interactive lessons and exercises. In accordance with current design principles, an object-oriented authoring system enabled the production of a quality CBT that meets the sponsor's budget and time constraints, and promises to be a key training asset for HH-60H personnel.

TABLE OF CONTENTS

I. INTRODUCTION 1

A. PURPOSE 2

B. COMMUNITY AND TRAINING BACKGROUND 3

C. PROBLEM STATEMENT 4

D. GENERAL APPROACH 4

E. SCOPE AND LIMITATIONS 6

II. PROBLEM BACKGROUND 7

A. THE ANVIS/HUD ENVIRONMENT 7
1. Night Operations 7
2. Use of Night Vision Goggles 8
3. Night Vision Goggle Training 11

B. ADAPTABILITY TO COMPUTER-BASED TRAINING 14
1. The Basic Instructional Systems Development (ISD) Process 14
2. Computer-based Training and Multimedia (MM) Development 16
3. Application Examples 23

III. METHODOLOGY 27

A. THE DEVELOPMENT PROCESS 27

B. SOURCES OF TECHNICAL ADVICE 31

C. HARDWARE AND SOFTWARE REQUIREMENTS 31

D. AUTHORING PROCEDURES 32

IV. RESULTS AND DISCUSSION 37

A. RESULTS 37
1. Changes, Enhancements, and Refinements 37
2. General Description of Final Version 38
3. Example Lesson 40

B. DISCUSSION 46

1. Composition and Limitations 47
2. Recommendations 48
3. Conclusion 49

REFERENCES 51

APPENDIX A. TRAINING REQUIREMENTS ANALYSIS 55

APPENDIX B. LAYER CONCEPT 63

APPENDIX C. CHANGES AND ENHANCEMENTS 65

APPENDIX D. REFINEMENTS 69

APPENDIX E. SCRIPT FOR MODULE 2, LESSON 1, PAGE 5 75

INITIAL DISTRIBUTION LIST 89

I. INTRODUCTION

The recent down-sizing within the Department of Defense has forced most fleet aviation components to operate with minimum resources. A prevailing working doctrine stating that "we must do more with less", dictates that aviation commands must meet readiness goals as efficiently as possible.

The Navy, along with the other services, has often relied on advanced technology to improve efficiency and force effectiveness. An underlying assumption is that our forces can use technology to their benefit in attaining readiness goals and improving mission performance. However, while technological advances may provide a potentially greater strategic advantage for the pilot, they also impose on the pilot a greater burden for understanding a wide variety of equipment and theory. In short, today's aircrews are under more pressure than ever to understand the constantly changing number of assets that are available to them; hence, the training systems that are delivered to the fleet must be efficient, cost effective, and as high a quality as the aircrews who use them.

Training aviators has become a highly visible task, and when this training involves night operations, the training procedures are scrutinized even more. In particular, the historical costs of neglecting night vision training, in terms of both personnel and material losses, demand more effective training methods. Fleet components may not always have the manpower to properly train all aircrew on night vision equipment. This being the case, either the quality of training suffers - and safety is sacrificed - or a better way to train is developed. Computer-based applications are quickly becoming the answer to many of the

fleet's training problems because (1) these applications provide a means to standardize training, and (2) they deliver needed training at the unit level.

A. PURPOSE

The Aviation Safety School at the Naval Postgraduate School is currently under contract by Naval Air Systems Command to develop a computer-based training application for the new AN/AVS-7 Aviators Night Vision Imaging System/Heads-Up Display (ANVIS/HUD) night vision equipment. This system is a combination of a night vision goggle (NVG) set and a heads-up display system that is ultimately going to be configured for installation in virtually all of the Navy's 1553 data-bus configured rotary-wing aircraft. Some computer-based instruction has already been developed for the UH-1N Huey helicopter trainer. This software provided a baseline for development of a computer-based training system that was designed specifically for the Helicopter Combat Support (HCS) community, which flies the HH-60H helicopter. The primary purpose of this research was to develop and deliver a computer-based application to NAVAIRSYSCOM, that provided high quality instruction and training on the ANVIS/HUD system to fleet HH-60H commands. The design, programming, and implementation was conducted under real-world deadline constraints. The secondary purpose of this thesis was to provide a thorough study of current computer-based instruction theory and multimedia authoring systems, and discuss how they can be applied to naval aviation training programs for night-intensive operations.

B. COMMUNITY AND TRAINING BACKGROUND

There are two HCS squadrons in the Navy; HCS-4, located at Norfolk, Virginia, and HCS-5, located at Point Mugu, California. HCS squadrons' primary missions are Special Warfare and Combat Search and Rescue. Both missions are similar, requiring HCS crews to train for operations in extremely hostile areas, often in the challenging night-time environment. Special Warfare involves the insertion or extraction of highly specialized teams (i.e. Navy SEALS or Army Rangers) in covert areas, while Combat Search and Rescue involves rescuing downed aviators or stranded service members from the battlefield or behind enemy lines. HCS squadrons prepare selected teams, or detachments, for action and squadrons always keep at least one detachment "at the ready" for immediate deployment. Detachments are extremely versatile and flexible, as they are designed to support operations afloat or ashore.

Squadrons are made up of roughly 30 percent Training and Administration of Reserves (TAR) personnel and 70 percent Selected Reserves (SELRES) personnel. TARs provide the stability in the squadron by working full work weeks, expediting administrative matters, coordinating deployments, and implementing training plans. On the other hand, in order to cut government costs, SELRESs are only required to be at the squadron for a short period of time. As a result of this unconventional attendance policy, training SELRES personnel can be difficult.

By design, HCS squadrons have more than adequate computer equipment, but are limited by personnel assets. For this reason alone, an HCS squadron is an ideal environment for a CBT program for the simple fact that instructors are not as accessible as

computers. In addition, all the Operational Flight Simulators and Weapon System Trainers for the HCS community are located at HS-10 - the Fleet Replacement Squadron in San Diego - which is not close to either of the HCS squadrons. Hence, both HCS-4 and HCS-5 must strive to conduct innovative on-site training either in the classroom, in the cockpit, or on a stationary computer.

C. PROBLEM STATEMENT

Provided with an existing ANVIS/HUD CBT for the UH-1N, what techniques and procedures should be utilized in converting the baseline CBT into a high-quality version that incorporates proper instruction design and requisite aircraft-specific changes for the HH-60H?

D. GENERAL APPROACH

PMA-205 provided the "baseline" interactive training CBT for the UH-1N, which was written using Asymetrix *Multimedia Toolbook (MTB)* version 3.0. Training on the baseline system took an average user approximately fours hours to complete, and was comprised of an overview module and four training modules. Modules were further broken into lessons, exercises and tests, and covered topics ranging from symbology to organizational maintenance of the ANVIS/HUD system.

While several different means of developing the HH-60H CBT were initially pursued using other possible applications, Asymetrix products were eventually chosen as the development medium. *MTB* is a multimedia authoring application that is based on

object-oriented design, using a structured programming language for use in a Windows environment. Ultimately, *MTB* was utilized to perform a series of re-engineering evolutions on the existing baseline system which are briefly described below.

From the onset, the development was broken into three "re-engineering phases". This approach was rather simple: at a minimum, a product needed to be produced and ready for delivery to Naval Air Systems Command that, if dictated by time constraints, might only incorporate the "bare essentials". Therefore, the first phase strictly involved converting all UH-1N data and graphics to HH-60H material. This first phase was subsequently broken into smaller steps:

- obtain and configure the requisite hardware and software.
- become familiar with the authoring system.
- establish initial liaison with the HCS community: Conduct interviews, obtain expert-opinion information, gather source material and graphics, and establish points of contact for future project team visits.
- formulate a needs analysis.
- develop or refine existing learning objectives and lesson plans.

The second phase dealt with making modifications in the system to better suit current computer-based instruction standards, and the final phase was intended to incorporate a limited number of multimedia enhancements as deemed appropriate. Chapter three addresses the software, hardware, and various modifications in detail.

E. SCOPE AND LIMITATIONS

The HH-60H ANVIS/HUD CBT is an excellent asset for the HCS community. However, as with all CBTs, there are limitations.

While the trainer provides excellent instruction without the presence of an instructor, the system is not designed as a stand-alone training device. It is designed to supplement training conducted in the ready-room and cockpit. In addition, any procedural concepts or principles that are published in the future will take precedence over what is taught in the trainer until the trainer can be updated to reflect the proper content and standards. Those responsible for follow-on revisions must keep this in mind.

When assessing the scope and limitations of the project in detail, the following points should be noted:

- Development for the actual HH-60H ANVIS/HUD is still in progress. Likewise, all publications related to the HH-60H ANVIS/HUD have not been completed; hence, source material was limited. What material that was acquired came in the form of facsimiles, preliminary copies of technical pubs, and notes from subject-matter experts.

- Because the equipment is still at the test level, hands-on experience with the actual equipment was also limited.

- Since there are very few military personnel that have actually flown with the current version of the HH-60H ANVIS/HUD equipment, subject-matter experts were scarce.

Despite these limitations, however, the CBT product that resulted from this research not only met, but exceeded the initial project objectives, and is ready for field testing.

II. PROBLEM BACKGROUND

A. THE ANVIS/HUD ENVIRONMENT

1. Night Operations

It is generally acknowledged that night operations present additional challenges, unique to the nighttime environment, above and beyond traditional daytime concerns. As a direct result of these challenges, today's military is faced with increased demands when conducting night operations. Therefore, it is strategically advantageous for a force to maintain dominant nighttime capabilities, effectively enabling them to capitalize on those additional demands also placed upon the enemy. Advantages enjoyed by the dominant night warrior are not solely linked to tactical surprise, but also include numerous added capabilities in areas such as logistics/supply, interdiction, strike, search and rescue, close air support, and tactical harassment achieved through around-the-clock operations.

America's first experiences with night air warfare came during World War II, primarily in the form of negative lessons learned from the Royal Air Force Bomber Command against Nazi Germany. For example, it was concluded that American forces lacked precision targeting capabilities. The limitations imposed on air power by a lack of night capability frustrated the American commanders in Korea and Vietnam just as much as it did their predecessors in World War II. In April of 1967, Admiral U.S. Grant Sharp, commander in chief of Pacific Command, formally identified the need to develop increased night targeting capability under adverse weather conditions as a top priority during the Vietnam War. This need was emphasized by Air Force Chief of Staff General John P.

McConnell that same month. In testimony before Congress, General McConnell stated that the Air Force was deficient in tactical air power, particularly in its ability to hit precision targets at night and in foul weather.

However, identifying a need is not sufficient to ensure a capability. By 1971, despite the best efforts of Admiral Sharp and General McConnell, the North Vietnamese continued to move 500 to 1,000 trucks *per night* down the Ho Chi Minh Trail with an average load of 8,000 pounds of cargo per truck. (McLean, 1992) The challenges of night air warfare have continued to increase and even in modern-day warfare there are considerable night operations, as evidenced by such conflicts in Libya, Iraq, Somalia, and Bosnia. American technology, training, and performance, have significantly advanced since the days of World War II, and pioneering advancements are continuing at an exponential rate. However, in order to take full advantage of technology as a "force multiplier," aircrews must be adequately trained to use new technologies in a safe and effective manner. The use of NVGs is a case in point, since the full benefit of NVGs depends so highly on aircrew understanding their capabilities and limitations.

2. Use of Night Vision Goggles

In order to determine training requirements for NVG use, it is necessary to understand aircrew mission performance needs for efficiently conducting sound nighttime operations. Combat operation survivability is directly dependent upon minimizing the threat's capabilities, while maximizing one's own. The ability to operate at night is a key strategy for reducing exposure to enemy forces and improving war fighting capabilities of our forces. The threat exposure reduction and element of surprise are achieved by

penetrating the battlefield in low-level flight under the cover of darkness. Hence, nighttime operations increase the tactical advantage of surprise and decrease the chance of detection, resulting in increased combat effectiveness. Operations at night are not possible in some cases without nighttime visual aids. On dark nights, the unaided human visual system is simply not capable of discerning visual cues that are necessary for visual contact, targeting, and terrain avoidance.

Therefore, nighttime aviation has placed increasing emphasis on night imaging devices that operate in the optical radiation portion of the electromagnetic spectrum. Night vision goggles are devices that amplify existing available light such as moonlight and starlight, to enhance nighttime vision. Through these devices, pilots acquire the capability to perform many daylight operations in the nighttime environment. However, to maximize the benefits of NVGs, aviation personnel require extensive training to acquire an understanding of the use, capabilities, and limitations of the goggles. (MAWTS-1, 1993)

To effectively exploit an enemy's vulnerability to night attack, aviators must first overcome the additional challenges presented by the night environment. They must not only be equipped with a capable night vision device, but must also be adequately trained in its use. Night vision devices generally combine both optical and electronic technologies, and are consequently referred to as electro-optics. The two most common electro-optic technologies are known as: thermal imagers, which incorporate infrared sensors, and image intensifiers, which exploit very low levels of natural illumination.

More limited roles of modern warfighting have highlighted the requirement to strike military targets, while minimizing the risk of damage to civilians and urban

infrastructure. Through the use of electro-optic targeting systems, United Nations forces in Bosnia, Iraq, and Somalia were able to attack hostile elements that had been deliberately placed close to or within urban areas and civilian population centers. These are important considerations for modern forces acting together under UN mandates. With the prevailing scrutiny of news media, the international political consensus and popular support needed for these military operations can be severely damaged by a single day's bad press. Offensive actions need to be carried out with minimum risk to friendly forces and civilian infrastructure. In modern military systems, it is electro-optics that provide the requisite minimum risk and maximum targeting options.

Today, electro-optic technology influences nearly all aspects of surveillance, observation, weapons targeting, and missile guidance. Because of the effective operating ranges of one to ten kilometers associated with electro-optics, its major impact has been realized in the conventional air-to-ground battle where combat ranges are well matched with effective ranges. Electro-optics provide the ability to see, detect, recognize, and engage targets with a speed, clarity, and precision that the unaided human visual system cannot. (O'Leary, 1995)

The AN/AVS-6 ANVIS is one form of NVG that is regarded as mission-essential equipment for night operations by many services. ANVIS is an electro-optical image intensifier system designed to provide aviators with the optimum capability to see in the dark and perform nap-of-the-earth and other terrain flight modes during starlight conditions. When integrated with a heads up display, the ANVIS combines to form the ANVIS/HUD AN/AVS-7 system which is designed to give aviators critical flight

information superimposed on the outside visual scan image of the ANVIS device. The composite system overlays cockpit information by projecting graphics onto either of the two monocle tubes used to display the night vision scene. (Troxel, 1993) It gives the pilot and copilot critical, real-time, high-resolution flight and navigational information, providing critical symbols including radar altimeter, attitude indicator, engine torque, master warning, and various waypoint indicators. By reducing the need to divert pilot attention from outside the aircraft to inside the aircraft to monitor flight and navigation instruments, ANVIS/HUD not only achieves its primary objectives of enhanced flight safety and reduced crew workload, but also offers additional advantages to increase operational effectiveness. The AN/AVS-7 was developed to reduce crew fatigue, maximize out-of-cockpit scan-time, and improve situational awareness.

3. Night Vision Goggle Training

Navy and Marine aviators flying at night with visual aids, such as NVGs, undergo extensive training related to the night environment including: characteristics and limitations of night vision devices, human factors associated with NVG limitations, and hazards of night flying operations. Two key training centers are currently in place to accommodate these requirements. Marine Aviation Weapons and Tactics Squadron One (MAWTS-1) essentially develops and disseminates NVG doctrine; whereas, the USAF Armstrong Labs provides training for instructors/instructor qualifications. To a certain degree, such instruction is based upon lessons learned from aircraft mishaps, as well as upon professional understanding about the performance of night vision devices based on limitations of current technology.

Present instruction in the U.S. military has improved over the years, given such innovations as the Night Imaging and Threat Evaluation Laboratories (NITE Labs) and development of more comprehensive and standardized instruction. In spite of these efforts, however, aviators continue to experience difficulties which sometimes result in mishaps. A review of recent Navy and Marine mishaps, for example, have shown that there have been no instances of goggle failure leading to an aircraft mishap.

Typically, mishaps that occur while goggles are being used are the result of the crew overestimating the capability of the night vision device and/or misjudging the associated risks of performing certain tasks in the hazardous flight regimes encountered. In some cases it is believed that aircrews would more readily perceive operational hazards, such a reduced visibility, impending Instrument Meteorological Conditions, or other situations that degrade NVG performance, if they had an opportunity to observe such situations in a training environment.

To some extent, NITE labs help expose aviators to potential hazards of low light intensity operations, in conjunction with various illumination and terrain variations. However, through the use of classroom video presentations and static Terrain Board demonstrations, it is difficult to address the vast range of dynamic situations that an aviator will encounter during changing environmental conditions. A major premise of NVG findings is that users need early, and continued, exposure to the night environment across a broad range of conditions in order to develop the perceptual skills needed to perform in the flight regimes required, and to enable them to respond in a timely and appropriate manner to changes in the visual environment.

The use of emerging technologies that enhance NITE lab instruction, such as computer-based multimedia presentations and advanced part task simulation, offer much potential in meeting the goal of enhancing aircrew NVG doctrine. By demonstrating goggle limitations, and most importantly, by providing a wide range of visual presentations that expose aircrew to visual phenomena likely to be encountered in the operational world, this goal can be realized.

Considering the unforgiving consequences of pilot in-flight errors, it is clearly a mistake for anyone to attempt to fly an aircraft using NVGs, if that individual's only NVG experience to date had been to don them for the pending flight. Substantial ground training must accompany any syllabus designed to prepare an aviator for flight operations utilizing NVGs. Regardless of the training program considered or the airframe for which NVG use is intended, the pilot must be educated in both the night environment in general, and the NVGs themselves. Training should include, but not be limited to, the following subject areas (Ciavarelli, Sengupta, and Baer, 1994):

Night Environment Analysis:
- Electromagnetic Spectrum Analysis
- Night Sky Illumination Analysis
- Terrain Variations and Effects
- Meteorology Analysis

NVG Analysis:
- History of NVGs
- Process of Night Vision
- NVG Limitations
- NVG Human Factors
- Generation of NVG-comparative Performance
- NVG Basic Operating Procedures
- Mount and Counterbalance Procedures
- Adjustment and Preflight Procedures
- Operation of NVG

Current NVG training is organized around classroom instruction, on-the-job training, on-site NITE Labs, and NVG compatible full-mission simulators (Ciavarelli et al., 1994). The associated safety concerns, which are a part of any NVG operations, demand that extensive training be conducted. Classroom training alone cannot convey the high degree of realism that is needed to appreciate the capabilities and the limitations of NVGs in order to improve safety. In the current environment of shrinking training funding, alternative methods of NVG training are needed. (Epperson, 1995) Interactive training systems and computer-based training need to be introduced on a much wider scale if scarce human resources are to be used in the optimum way, both in the military and civil fields. It is maintained that CBTs are capable of reducing training time on real equipment, without reducing training benefit. (Lok, 1989)

B. ADAPTABILITY TO COMPUTER-BASED TRAINING

1. The Basic Instructional Systems Development (ISD) Process

The successful design of any training system must be evaluated in terms of the desired learning objectives. Will the student attain the knowledge and/or skill(s) intended, and needed, to perform capably on the job? The designer must clearly establish desired learning outcomes at the outset so that the training process may develop requisite knowledge and skills in the student. The ISD approach combines traditional educational and technological approaches to emphasize the identification of student needs, so as to achieve desired competency upon completion. It is a systems approach to making a rational selection of valid training content, materials, strategy, and media to meet job-

performance requirements (Larsen, Randle, and Popish, 1994). This ISD approach is briefly summarized in the five phases of Figure 2.1 and the following phase descriptions:

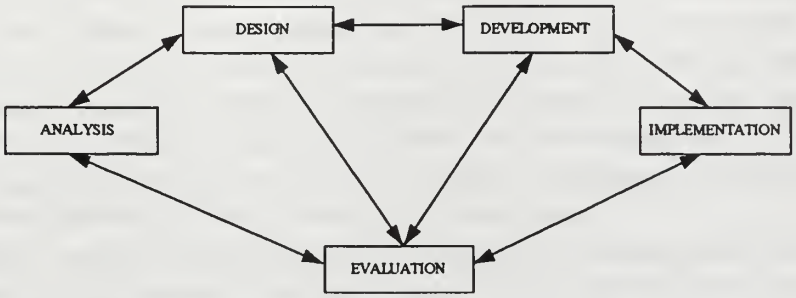

Figure 2.1. Major Phases of ISD (After Kearsley, p.105, 1983)

- *Analysis*: The analysis phase establishes the need for training, and identifies each of the specific training demands. The analysis also satisfies the technology requirement through a comparison of the proposed system, for which the training is being developed, with that of the existing system. An organizational point of contact should also be identified in this phase to ensure that someone at the organization remains better trained on the system than the students themselves.

- *Design*: Defining the learning objectives and classifying each as either knowledge or skill, is the foundation to successful ISD. Performance standards should be established to ensure that the correct steps are taken, and in the proper order. The student should be profiled in terms of level of expertise (expected material knowledge, education, computer literacy, etc.). It is in this design phase that curriculum outlines and lessons are developed, and media/system options are considered.

- *Development*: Development deals with the actual presentation of instruction, tailoring a different instructional strategy for each of the different learning objectives. For instance, instructional strategies to facilitate teaching conceptual tasks differ from those for procedural tasks, in terms of what will be presented to the student and what will subsequently be expected.

- *Implementation*: The implementation phase must give due consideration for staff training, verification of the administrative framework, quality control measures, system installation, and on-site testing.

- *Evaluation*: An effective evaluation should not only be *formative*, but should be *summative* as well. That is, it should consider system evaluation both before and after delivery. Optimally speaking, the evaluation should be continuous and

should incorporate pertinent test and evaluation methods, data collection, and recommendations for the improvement process. (Ciavarelli et al., 1994)

2. Computer-based Training and Multimedia (MM) Development

When considering the history of computer-based training and multimedia development over the past 35 years since inception, it helps to formally conceptualize the particular process being described. Computer-based training addresses that instructional discipline which invokes the use of computers for educational purposes. What was once strictly a government-funded venture carried out through the use of mainframe and minicomputers, is now achievable on microcomputers, and is a common corporate tool tailored for industry, the trades, school systems, and even individuals.

Multimedia, on the other hand, is a term commonly used to describe the simultaneous integration of multiple forms of media, such as text, voice, music, animation, video, graphics, and other forms of data. The integration of various media forms is intended to enhance the role of the computer as a communications device, in this case, as an instructional platform.

A comprehensive review of instruction design principles was presented by Ciavarelli et al. (1994), and will not be repeated here. This review was based on prevailing theories of instruction following the works of Gagne and Briggs (1979), Mager (1984), and Merrill (1983; 1980). By way of summary, these instruction design experts maintain that the quality of instruction, whether delivered by computer or not, is dependent upon key principles of design which include (1) careful specification of learning objectives, (2) formation of instructional strategies for the delivery of instruction based on

the content and performance requirements specified in the learning objectives, (3) provisions for adequate student guidance and organization of instruction to promote learning, and (4) evaluation of student progress and instructional effectiveness, using tests based explicitly on the stated learning objectives. If any of these particular instruction design elements are missed during development, or are missing in the instruction, the quality and learning value of a course is seriously compromised.

The history of computer-based training is, unfortunately, filled with instances of poor design and implementation problems. Kearsley, Hunter, and Seidel (1983), and Montague, Willis, and Wulfeck (1983) reviewed the history of early efforts to develop high-quality instruction delivered by computer, and reported some of the lessons learned. Some of the most common pitfalls of poor design and implementation of computer-based instruction are summarized in the following paragraph.

Known failures of CBT/MM have commonly been attributed to breakdowns in one or more of various instructional design principles. *Learning objectives* are often not defined, or are badly conceived, and/or poorly written. *Testing* is frequently administered in a manner inconsistent with stated objectives in terms of both content and performance. Quite probably the most expansive category of CBT/MM criticism is found in its *presentation.* Problem areas with presentation include insufficient *interactivity*, inadequate *materials*, *text bias* (text overload resulting in a CBT which is merely an electronic page turner), lack of *practice* opportunity prior to testing, poor human *interfaces*, and insufficient transferability of learned skills to actual domains of use.

In order to improve instructional quality, specifically for CBT, some authors have published design and development guidelines which provide advice at each stage of CBT development. One of the most useful sources providing such guidelines is *Computer-based Instruction: Methods and Development* by Alessi and Trollip (1991), which presents a step-by-step process of CBT development including instruction design principles, development templates such as story boards, and guidelines for computer screen layout. Kearsley (1983) briefly reviews some of the common flaws of CBT, and provides general advice on how to improve instructional presentations. Another, more detailed analysis of design and development procedures and prescriptions for improving CBT instruction, is delivered by Gery (1987). Her presentation is substantially benefited by the author's extensive experience with both good and bad examples of CBT. Finally, in the special case of multimedia technology application, Oblinger (1992) specifically focuses on different kinds of media (audio, video, graphics, animation), and the appropriate application of each in CBT.

The interface, in particular, is cause for much concern regarding presentation quality. No matter how good the instruction content is in CBT, if the student has difficulty using the computer and interacting with the presentation, then potential value of instruction is damaged. Over the years, human factors specialists have identified some of the common flaws associated with poor human interface designs. Some of the most prevalent problems involve screen presentation formats and navigation between various screens. (Ciavarelli, et al., 1994) CBT/MM applications often do not effectively describe system functionality to help the user understand and operate it. Poor design also results

from a failure, on the part of the human interface, to reflect the form and level of user guidance considered most appropriate. As a result of such design deficiencies, human factors specialists have also published various guidelines to help improve human-computer-interfaced design. One useful source is Jones (1988) who outlines some general design principles for screen format, navigation rules, color selection, and such human engineering considerations as character size and legibility. A few of his suggested guidelines are itemized below:

Human-Computer Interface Screen Guidelines:

- Maintain consistency in display format, information layout, and position.
- Use landmarks and signposts to show current location, path traversed, and what lies ahead.
- Indicate present operating mode.
- Indicate start and completion of each task.
- Give the user a way out of a predicament.
- Control display density by logical spacing and layout.
- Maintain page uniformity.
- For text, use both upper and lower case, and vary font and size only for emphasis. Standard font size should be no smaller than seven points. Avoid red and blue colors for text.

Having discussed, in some detail, common pitfalls of CBT/MM, it should be realized that the ANVIS/HUD Trainer endeavors not to repeat these design failures. Care has been taken to verify that design is based on sound practices and usability is maximized.

To fully exploit the capabilities of CBT/MM as part of the ISD approach, a thorough understanding of the learning process is imperative. Although some lessons seem simple to learn, the act of learning is no simple endeavor. Learning involves a complex, interrelated series of cognitive processes, including attention, perception, and memory. Based on *cognitive psychology*, the science of how people process information, the principles of instructional design can help designers create teaching and training materials that are consistent with the way people learn. (Clark, 1995)

By comparison, the teaching and learning potential of CBT/MM is not necessarily better or worse than that of the traditional classroom, provided that both incorporate the same design methods. It should be emphasized that these *methods* are the instructional techniques that facilitate learning and guide the use of a particular *medium*. By contrast, the media are the means of implementing those methods. "Any medium can be rendered ineffective by inappropriate methods (Clark, 1995)." Likewise, independent lessons designed using similar instructional methods, but presented via different media, yield similar results. A tedious classroom lecture where students are inundated with information, yet given no opportunity to work with it, produces similarly poor results as the CBT/MM program that does not allow for adequate student interactions. The ineffectiveness in either case is attributed to learner overload, with inadequate opportunity to develop skills or practice content. What makes the CBT/MM environment so well suited as a successful media type, however, is the ease with which it satisfies common methodology requirements, through the of demonstrations, animations, examples, practice, feedback, and remediation.

The CBT/MM designer's understanding of the learning process is once again emphasized as crucial to effective implementation of the previously described instructional methods - those psychologically active elements of teaching, training, and learning. Perceived information, detected through the senses, is temporarily stored in the working memory of the brain. In order for the information to actually be learned, however, it must somehow be transferred from working memory to long-term memory. Working memory is short-term memory which describes the conscious center of the brain and is storage-capacity limited. This limitation highlights the need for moving knowledge and skills from the working memory into long-term permanent memory.

In order to avoid loss of information from the working memory of the brain, that information must be practiced, or *rehearsed*, in some way. Most rehearsal occurs in the form of image visualization, or other reorganization of the information in one's mind, to subsequently be remembered. Effective rehearsal successfully captures information, or *encodes* it, in long-term memory - thereby allowing for the transformation of that information to *knowledge*.

The ability to store information on a long-term basis is recognized as a key goal of effective teaching and training, but the final goal is to provide a means for retrieval of this information. Robust instructional methods are clearly essential to promote effective human information processing at both of these levels. The effective management of cognitive load - the amount of information that an individual is capable of processing at one time - can be realized as a result of lessons founded on sound instructional methods.

Considering the goals of effective teaching and training, the following instructional methods are recommended as CBT/MM guidelines to avoid overloading the learner's working memory, encourage encoding from working memory into long-term memory, and help ensure the learner's ability to retrieve knowledge when desired.

***Summary of Instructional Strategies** (Clark, 1995, pp. 26-27):*

- *Simplicity*: Keep cognitive load low with simple screen designs and sparing use of text, sound, motion, color, etc.

- *Modality Congruence*: To avoid dividing the learner's attention, use various media elements such as text, graphics, and sound to present reinforcing rather than disparate messages.

- *Cueing*: To direct learner's attention to important parts of the message, use color, arrows, shading, and sound sparingly.

- *Adjunct Memory Support*: Keep visible on the screen the information the learner needs to refer to during the instruction, especially to respond to questions.

- *Frequent Rehearsal*: Clear working memory by encouraging frequent rehearsal which will move information into long-term memory.

- *Concrete Words and Reinforcing Modes*: Encourage dual encoding through the use of concrete words and different modes to reinforce messages.

- *Promotion of Elaborative Rehearsal*: Avoid rote repetition in interactions. Instead, design interactions that match job activities and skills.

- *High-fidelity Simulations for Procedural Skills*: For procedural skills, encourage encoding specificity through the use of high-fidelity simulation practice. Simulations should replicate the actual job environment as closely as possible.

In terms of overall effectiveness of a presentation, multimedia traditionally invokes a perception of gained power, motivation, and captivation of audience. There is, however, a real danger in succumbing to blind faith in this notion. Not only does multimedia afford the designer the capability to create powerful, awe-inspiring presentations, but it gives the

opportunity to barrage users with overwhelming amounts of superfluous information. Designers must be conscious of these capabilities and associated pitfalls to successfully design effective systems, by adding such enhancements only where improvement in instructional value and/or student motivation is benefited.

3. Application Examples

As with the implementation of any training program, there is no guarantee of success. Effectiveness can only be achieved through careful planning, astute design, and thorough scrutiny of past successes and failures of similar programs. Crucial lessons learned should be identified and incorporated. With this in mind, the ANVIS/HUD Trainer demonstrates significant potential and its success is firmly supported by the achievements of previous comparable aviation-related CBTs. A brief account of three separate CBT success stories appear in the following paragraphs.

To become fully qualified in the AH-64A attack helicopter, a student aviator must learn to identify and interpret the individual symbols presented on the helicopter's visual displays and to interpret information provided by groups of symbols. A "Symbology Tutor" was developed to achieve these objectives. Although specific results of training module implementation are not known, design was based to take advantage of the following benefits: self-instructional format not requiring an instructor, capability of providing immediate feedback and remedial instruction, and suitability for both skill acquisition training in an institutional setting and skill sustainment training in an operational setting. (Ruffner, Coker, and Weeter, 1989)

The F/A-18 Pilot Training System has successfully employed computer-assisted instruction (CAI) since operation commenced in August, 1982. Not only do statistics regarding safety record and pilot throughput attest to training effectiveness, but the students, themselves, unconditionally stated that they were better prepared to fly the F/A-18 than any other aircraft. CAI is used to present both cognitive and procedural types of information. This versatile use of CAI saves valuable instructor and simulator time, and greatly enhances student's progress in the F/A-18 program. Other advantages realized by this self-paced, one-on-one program occur in the form of accelerated speed, improved quality, immediate feedback and remediation, and ease of modification. (Williams, 1984)

Regarding the procurement of the Army LH-class helicopter, simulation has played a vital role in cost control and design optimization. To avert the drain on valuable simulator time, a desk-top training station was built to emulate the capabilities of the crew station in the full-mission simulator. The trainee's view of a tutorial is that of a self-paced demonstration of system operation and procedures. Performance feedback reveals improvements in both speed and accuracy. The overall goal of the desk-top trainer as a stand-alone training system was to unburden the full-mission simulation facility, so that training could be effected for at least 75 percent of the systems available in the LH-class helicopter. This goal was not only met, but exceeded, and accomplished in a timely and standardized manner. (Becker, Matsumoto, Phillips, and Kennedy, 1990)

With success stories such as these paving the way, there is considerable justification for ANVIS/HUD Trainer potential. Fundamental design principles characteristic of these proven training programs have been consciously integrated into the

ANVIS/HUD CBT, functionally enhancing it as an effective training mechanism. The methodology followed to accomplish this is discussed in Chapter III.

III. METHODOLOGY

The steps taken to develop this training product represent a combination of the ISD process outlined in Chapter II and a similar process known as the Systems Development Life Cycle (SDLC) (Whitten, 1994). Both the ISD and SDLC processes are based on the same general concept that most developmental projects involve a series of cycles and stages. While minor deviations exist throughout various industries, the SDLC is a common systematic approach to solving business problems and developing information systems. In keeping with the concept that the end product of this research was not only supposed to be this thesis, but more importantly was supposed to produce a usable "information system" for the Department of the Navy, an ISD/SDLC variant was devised and implemented to develop and re-engineer the HUD training system.

A. THE DEVELOPMENT PROCESS

Since time constraints dictated the project's development schedule, several of the customary procedures and guidelines found within each stage had to be condensed or even completed in parallel - vice the "typical" sequential flow of execution discussed in Chapter II. The implementation and evaluation stages are addressed in Chapter IV. The following A brief description of the development process follows:

1. Analysis: This phase primarily consisted of analyzing the UH-1N baseline system, and determining how well it met the basic ISD and CBT principles in terms of

navigation, color selection, screen clutter, page uniformity, etc. The following observations were made:

- The baseline system was created with *Asymetrix Toolbook (MTB)*, Version 3.0. *MTB* products, like most authoring systems, are based on object-oriented design (OOD). *MTB* applications are programmed in a language called "OpenScript" which is similar to PASCAL.

- Code for the majority of the baseline product was embedded, and attempting to gain access to and printing the numerous files was initially not possible. After two weeks of application familiarization and technical assistance from *MTB* technical representatives, access was gained to *most* of the code on the baseline system.

- Designers of the baseline did an exceptional job in data organization, presentation, and student remediation. In addition, when the baseline was compared to the CBT lessons-learned outlined in Chapter II, learning objectives, tests, and presentation format were also well designed.

2. Design: This phase primarily consisted of CBT/MM authoring system selection and orientation, and equipment familiarization. The following observations were made during this phase:

- There are three leading authoring systems on the market today: Asymetrix's *MTB*, Macromedia's *Authorware*, and Aimtech's *IconAuthor*. In terms of multimedia applications, many professionals within the industry argue in favor of both *IconAuthor* and *Authorware* when compared to *MTB* applications.

- It was discovered that the original system was designed by a full-time government-contracted development team. Since the baseline system was designed by a team of professional programmers and analysts, efforts were made to take advantage of the team's expertise.

- *MTB* released a CBT version that was well suited for the purpose of this research. Hence, the design phase focused on capitalizing on the assets of the baseline system to produce a useful second-generation system for the HH-60H community. *MTB* CBT Version 4.0 was chosen as the authoring application.

- Since the equipment is still undergoing testing, acquiring a set of AN/AVS-7 devices proved impossible. Equipment familiarization was conducted through publications and technical consultations.

3. Development: Since this was evolving into a re-engineering project, changes were prioritized and broken into "must have, should have, and nice to have" categories that took the form of revisions. (Clemons, 1991) With traditional design, both development and re-engineering processes take on life cycles similar to the one seen in Figure 3.1.

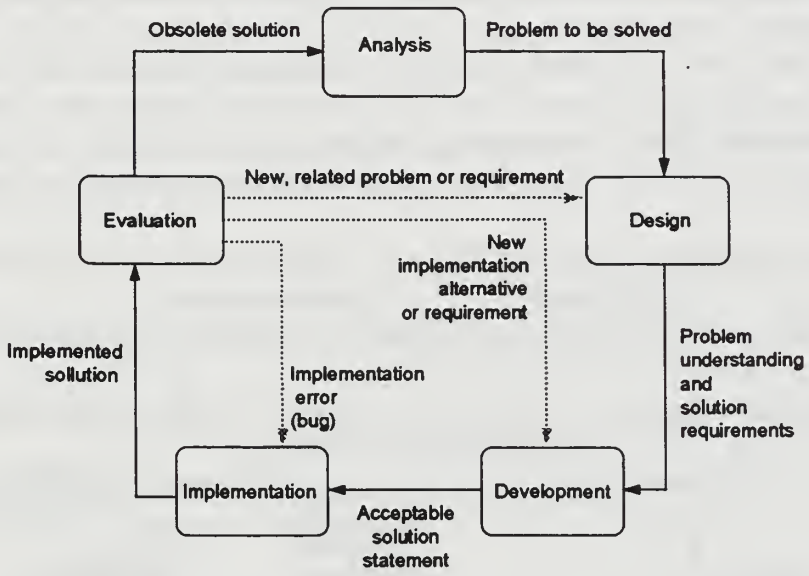

Figure 3.1. SDLC Process (Whitten, 1994)

Due to the time constraints, however, design and analysis were conducted nearly concurrently through revision processes. The following revisions were planned:

- Revision 1 was to isolate required changes - making the absolute minimum changes to convert the UH-1N product into a HH-60H product.

- Revision 2 was to incorporate graphic improvements, more robust handlers, and rectify any erroneous procedures and "bugs" found in the baseline system.

- Revision 3, the final version, was designed to incorporate any multimedia enhancements, but only if they were to improve the training value.

As a result of this revision process, modified loops formed. This loop concept proved conducive to the team approach of thesis research: while one member of the team identified the required changes through communication with points of contact, the other member identified erroneous procedures in the UH-1N baseline while concurrently designing the HH-60H version based on the other member's findings. The process of updating and modifying the learning objectives for each lesson - normally performed in the design phase - was also concurrently conducted. Learning Objectives were defined and published in Ciavarelli (1996). The resulting data flow appears in Figure 3.2 below.

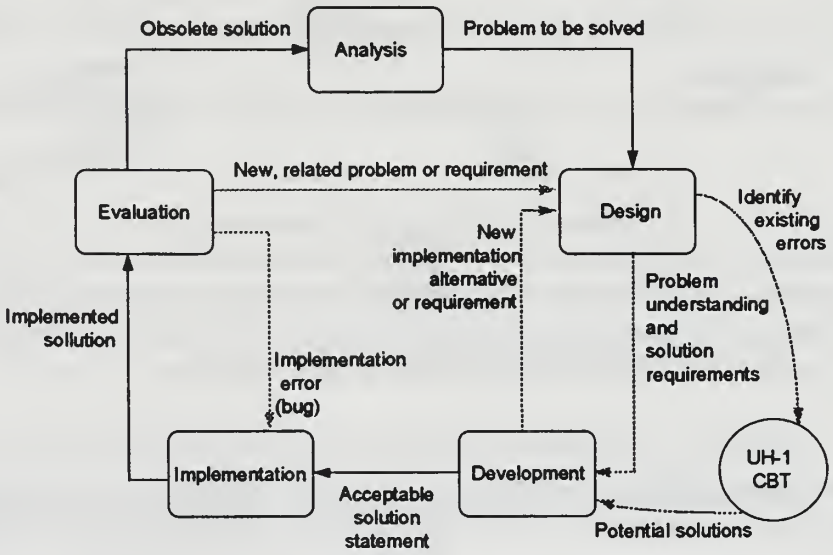

Figure 3.2. Modified SDLC Process (from Whitten, 1994)

B. SOURCES OF TECHNICAL ADVICE

Several points-of-contact served as primary sources of technical material and expertise. Contacts ranged from the Project Manager at Naval Air Systems Command to local personnel familiar with the ANVIS/HUD project. These contacts provided the requisite manuals, publications, and notes through mail, phone, and facsimiles.

In addition to these devices, a trip was made to HCS-5 at Point Mugu. While establishing initial liaison between the HCS community and the Naval Postgraduate School ANVIS/HUD project teams, the trip also served as an information gathering venture. During this trip, the project's needs analysis was refined through interviewing subject matter experts on mission planning, training, equipment, and crew composition. In addition, literature and documentation for ANVIS/HUD symbology and training procedures were acquired. Appendix A contains an example of the needs analysis form that was utilized during the Point Mugu trip, as well as a summary of the information gathered from the various subject matter experts of HCS-5.

C. HARDWARE AND SOFTWARE REQUIREMENTS

The trip to Pt. Mugu had a great impact in determining the software and hardware resources needed for development. After discussing the training environment with future users, it was ascertained that video and sound enhancements were neither required or desired. As a result, the following software and hardware were used for development:

- Pentium® 100 PC with 4X CD-ROM
- Shamrock II Super VGA Monitor
- HP Laserjet II Laser Printer
- Standard Video Card

- HP Scanjet IIc Color Scanner
- Asymetrix Multimedia Toolbook Authoring Application, Version 3.0
- Asymetrix Multimedia Toolbook Authoring Application, CBT Version 4.0
- Aldus PhotoStyler Graphics Editor version 2.0
- MS Paintbrush
- MS Powerpoint

In addition to the hardware and software design requirements, per the sponsor's requirements (NCCOSC, 1994), the ANVIS/HUD training system was *proposed* for operation for use on the following system configuration:

- Processor: 386/25 MHz
- Hard Disk: 100MB
- RAM: 4MB
- Graphics: VGA
- Mouse: Microsoft Compatible

D. AUTHORING PROCEDURES

The following section details the procedures used in authoring and revising the HH-60H ANVIS/HUD CBT:

Step 1: *Categorize the changes required.* There were three major types of changes that needed to be made to the baseline system: (1) changes that required text modifications; (2) changes that required graphic modifications; and (3) changes that involved not only text and graphic modifications, but also included coding modifications.

Step 2: *Separate the hard from the easy, and attack the easiest changes first.* Modules one, three, and the majority of four required very little coding changes. Hence, only one third of the authoring procedure time was allocated to modifying these three modules.

Step 3: *Determine the highest object-level at which a change can be made.* Asymetrix *Multimedia Toolbook* utilizes a "Toolbook" or object-oriented design concept. Any applications made using *MTB* are based on a "master" book, with the master book containing global variables, handlers and commands. From the master book, smaller books, or "mini-books", are designed to handle more specific tasks. Likewise, each book is further broken down into "pages". Each page is finally broken down into the least common denominator - objects. These objects can be defined in various methods: as field objects (text), paint objects (graphic forms), group objects (composite graphics), etc. Each object, whether it be a group object or a simple dot within that group is assigned a unique identification tag, and the objects are arranged on a page in layers. Appendix B shows an example of *MTB's* "layer" concept as it applies to the baseline system.

Determining the highest level that a change can be made to an object greatly reduces the workload required. For example, if an object is declared at the "book" level, it is considered global and could appear on every page within that book. Hence, changing that object at the global level requires only one modification, and precludes the designer from having to go into every page and changing each object one at a time.

Step 4: *Repeat step 3 as necessary.* By using the object-oriented design concept and the revision process mentioned above, a "hierarchy" approach was developed. Once all the changes were made at the "book" level, then the lessons of each book were inspected and changed as necessary. Once each lesson was changed, each page was inspected, and so forth. Once all the lessons within a book were completed, the process

started over again on the next sequential book until all the "easier" modules were completed.

Step 5: *Attack the hard.* Modules two and a portion of four were interactive and required substantial coding changes. Hence, these areas were allocated the remaining two-thirds of the authoring time. Fortunately, just like the simpler changes, code changes could be implemented in a similar fashion - by determining the highest level, implementing all the applicable changes, and working down the hierarchy.

Figures 3.3 and 3.4 show a simplistic "before and after" example of the changes

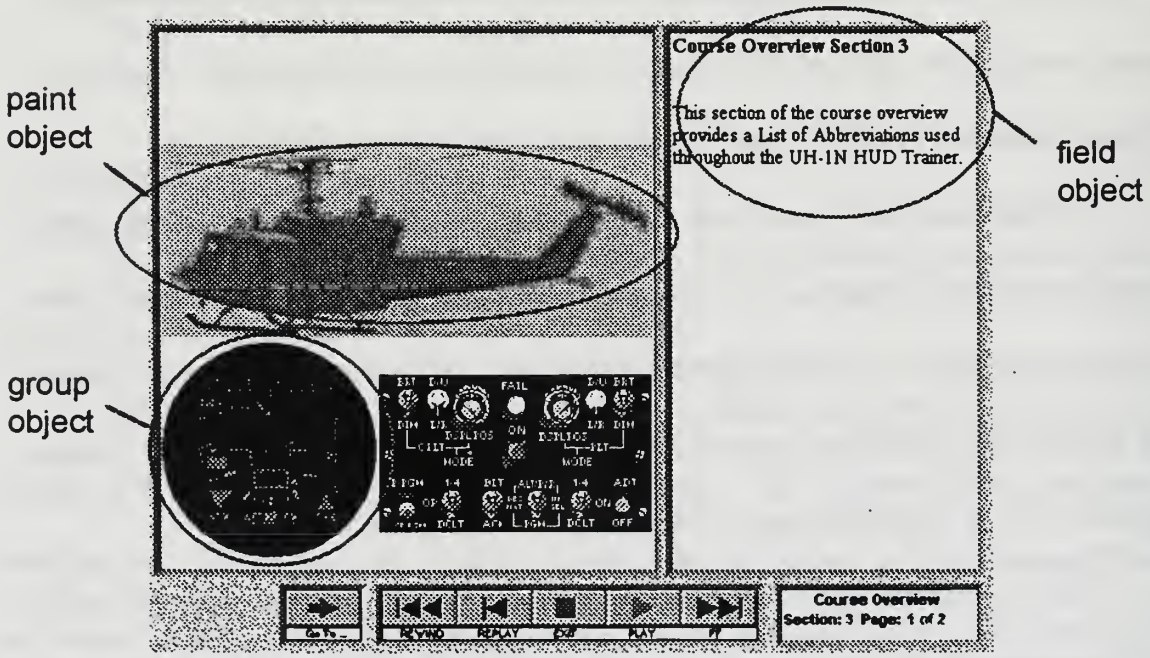
paint object
group object
field object
Course Overview Section 3
This section of the course overview provides a List of Abbreviations used throughout the UH-1N HUD Trainer.
Course Overview
Section: 3 Page: 1 of 2

Figure 3.3 UH-1N Overview, Section 3 Page 1 taken from the original system

that were made to just one of over 500 pages within the project. The example includes group, field, and graphic object changes. While this example page reveals only a few of the

types of changes that were required, Chapter IV goes into greater detail about the results of the authoring and revision processes.

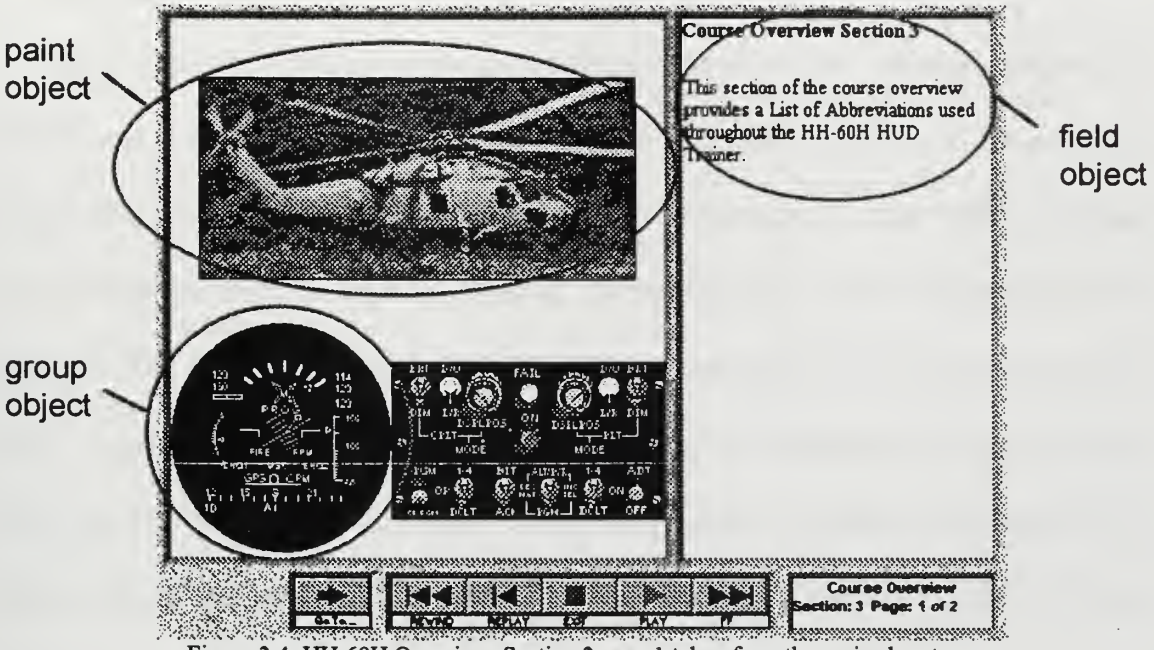
paint object
group object
field object
Course Overview Section 3
This section of the course overview provides a List of Abbreviations used throughout the HH-60H HUD Trainer.
Course Overview
Section: 3 Page: 1 of 2

Figure 3.4 HH-60H Overview, Section 3 page 1 taken from the revised system

IV. RESULTS AND DISCUSSION

A. RESULTS

The final product resulted in the successful revision of all five modules. Because the UH-1N and HH-60H ANVIS/HUD systems are markedly different, it comes as no surprise that the revision process outlined in Chapter III produced a final CBT version that was substantially different from the original. While the casual observer may mistake the HH-60H CBT to be similar to the UH-1N CBT, closer inspection reveals a system that differs in graphic applications, programming execution, and interface handling throughout essentially every module. This chapter reviews the resulting changes, enhancements, and refinements; outlines the system architecture; and describes a selected lesson of the final product.

1. Changes, Enhancements, and Refinements

In accordance with the "80-20 rule" of program re-engineering (Abdel-Hamid and Madnick, 1989), eighty percent of the effort was required to produce a mere twenty percent of the system. While the overview and modules one, three, and the majority of four largely required graphic creation and object modifications, module two and the last lesson of module four involved extensive program modifications and improved handlers. Consequently, modules two and four required most of the effort, although they account for only 18 percent of the CBT's bulk. The resulting modifications can be categorized into three groups:

- Changes - The result of *replacing* UH-1N graphics and text with HH-60H graphics and text;

- Enhancements - The result of taking existing baseline code and/or handlers and *improving* them - but *not* replacing them;

- Refinements - The result of *correcting* erroneous text and code that existed in the baseline system. While the original UH-1N system was well designed, there are areas throughout the original system that do not have any navigational features, contained erroneous handlers, or produced incorrect results in the exercises and tests.

Results of the multiple changes and enhancements introduced in Chapter III are grouped together and summarized in Appendix C. There were many changes made in each module that were global changes - simple text or graphic changes that had to be incorporated into every module. Since many of these changes were similar, including each of them in the Appendix would be mundane and excessive; hence, only selected modifications and improvements are highlighted. Appendix D lists the refinements of each module. As opposed to Appendix C, however, Appendix D lists every significant correction *in detail* for the purpose of documenting results for possible follow-on efforts.

2. General Description of Final Version

The final product's structure resembles that of the original: one introductory module and four training modules. Each module contains at least three lessons, several exercises, and a test to ensure adequate coverage of the material and evaluation of the student. A brief description of each module follows:

- Overview Module: Describes the structure of the trainer, demonstrates the Video Cassette Recorder VCR-style of navigation, and provides examples of how to answer exercise and test questions. In addition, this module contains a library of all ANVIS/HUD related acronyms, cautions, and warnings.

- Module 1: HUD System Overview: Introduces the student to the various HUD components including the locations and functions of the display unit, signal data converter, converter control, collective stick switches, and the built-in-test features.

- Module 2: HUD Symbology: An interactive module which introduces the student to the limitations and features of the 31 symbols of the HUD display, describes the procedures for programming the HUD symbols, and demonstrates the procedures for calibrating the pitch and roll indicators.

- Module 3: HUD Operations: Introduces the student to basic maintenance and handling techniques of the sensitive visual equipment and lenses, discusses HUD start-up procedures, and covers egress procedures using the HUD system.

- Module 4: Organizational Maintenance: The largest module, it covers the hazards of electrostatic discharge, and it provides information on HUD shipping, fault isolation, and weapons replaceable assemblies. In addition, the final lesson is interactive, presenting a troubleshooting matrix to the student that covers a host of potential equipment-related malfunctions.

Each module is broken down to the smallest level of detail required to best present the information. Figure 4.1 shows a block diagram of the trainer's structure, and a flow through Module 1 has been arbitrarily highlighted for illustration.

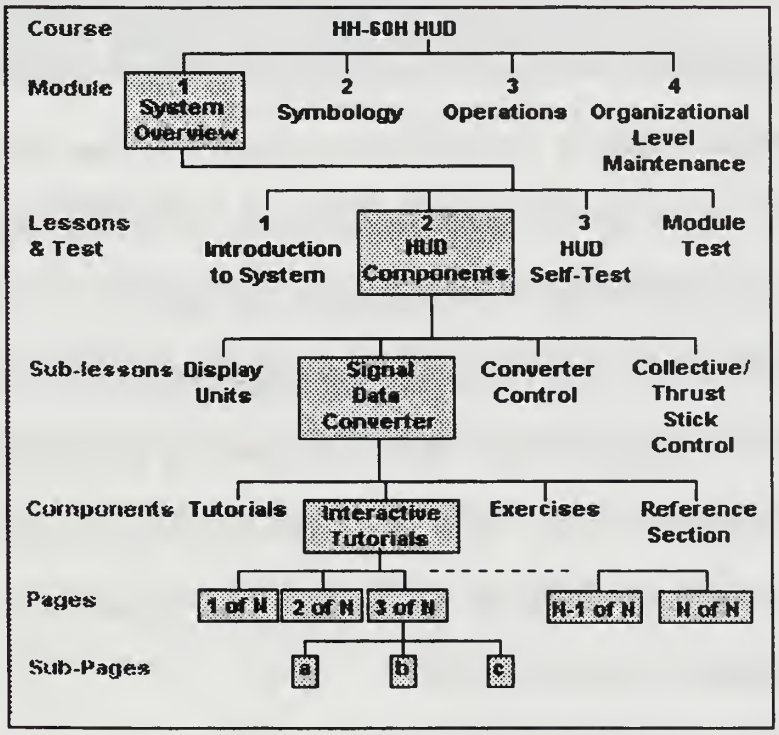

Figure 4.1. ANVIS/HUD CBT Structure

3. Example Lesson

While virtually countless interrelated operations take place throughout the aforementioned lessons of the trainer, the following is included as one isolated example of the coding and inter-activity contained in the CBT. One should keep in mind that the graphics, handlers, and programming associated with this example pertain to only one page out of over 500 total pages. While the coding may be shared in various areas throughout the CBT, those occurrences are limited.

The page shown in Figure 4.2 is taken from module 2, lesson 1. In this lesson, the student is introduced to all 31 of the many symbols that appear on the ANVIS/HUD display and the student is informed of the various characteristics of each symbol. This lesson can be either sequential or entirely interactive, depending on the student's method of navigation. The student can choose to navigate utilizing the traditional VCR-style buttons, or choose to navigate by moving his cursor around the screen. When the student places the cursor over a specific symbol and clicks the left mouse button, the CBT navigates to the appropriate page corresponding to that symbol - at which point the student can read about the correct terminology and associated specifics of that particular symbol. Figures 4.3 through 4.8 show excerpts from areas of interest within the script that pertain to the referenced page. Again, this is only one lesson of many that require interaction from the student. Subsequent lessons require the student to isolate, delete, or even move these symbols around the screen.

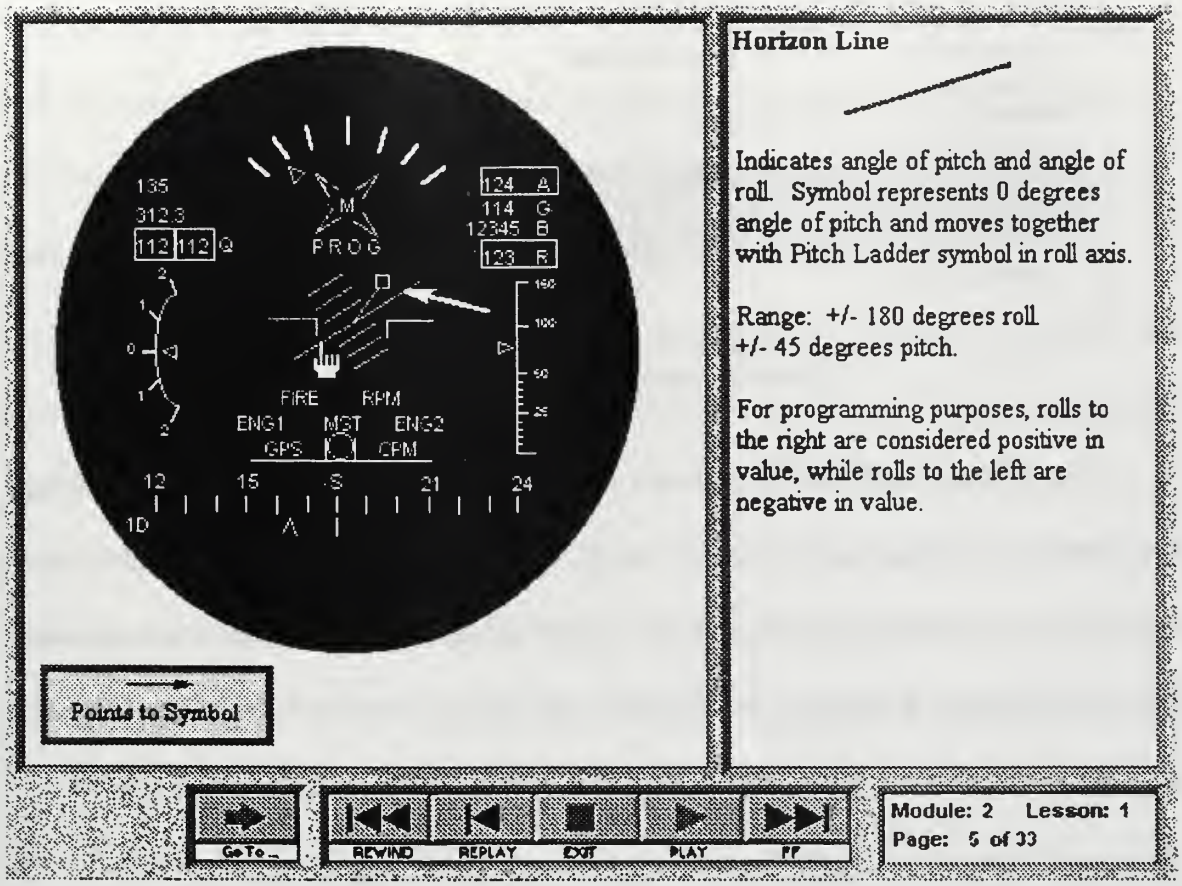

Figure 4.2. Module 2, Lesson 1, p. 5

The symbol discussed in the display above is the "Horizon Line". Notice the arrow pointing to the symbol, and the cursor location represented by the "hand" figure. When the student places the cursor over a symbol and clicks the left mouse button, the trainer will access one of several arrays that will automatically call up an arrow and the associated text that describes the symbol of interest. If the student chooses not to navigate via the cursor control, the VCR buttons are still active, and can be utilized to browse each symbol sequentially, as it is declared in the array. While multi-dimensional arrays are used throughout the trainer, Figure 4.3 shows the Horizon Line declaration which required only a one-dimensional array.

```
...
svNameSymbol[3] = "Aircraft Fixed Reference"
SymbolIndex[3] = 3
svNameSymbol[4] = "Pitch Ladder"
SymbolIndex[4] = 4
svNameSymbol[5] = "Horizon Line"
SymbolIndex[5] = 6
svNameSymbol[6] = "Angle of Roll Scale"
SymbolIndex[6] = 7
...
```

Figure 4.3. Array symbology declaration

The arrows are declared in a similar fashion, with start and end points correlating to an inverted Cartesian plot (an X-Y graph oriented up-side down), with each axis ranging from 0 to 8000 units. Each time a particular symbol is selected, the associated arrow is displayed as a corroborative device for the student to verify his selection. Figure 4.4 shows the declaration of the arrow associated with the Horizon Line symbol.

```
...
svLineStartX[5] = 2685 --        Pitch Ladder
svLineStartY[5] = 2440
svLineEndX[5] = 1995
svLineEndY[5] = 2220

svLineStartX[6] = 3255 --        Horizon Line
svLineStartY[6] = 2340
svLineEndX[6] = 3720
svLineEndY[6] = 2460

svLineStartX[7] = 3270 --        Angle of Roll Scale
svLineStartY[7] = 1290
svLineEndX[7] = 3630
svLineEndY[7] = 1770
...
```

Figure 4.4. Arrow declaration

One can see that the number within the brackets correlates with the symbol's index declared in Figure 4.3. In the case of the Horizon Line, the page that correlates to the

symbol is "[5]", and the symbol's index is "6". As a navigation example, if the student was to navigate via the VCR controls, as he paged forward from page four to page five the associated objects that correspond with the Horizon Line would appear because the Horizon Line's "svNameSymbol" declaration corresponds with page 5.

As previously mentioned, this module is interactive in that it will jump or "hyperlink" to the correct page when the user selects a particular symbol. Since *MTB* incorporates a "hyperlink" feature, this navigation process is executed through the script by declaring "hot-spots". If the user places his cursor within the boundaries of a hot-spot and clicks the mouse button, the program accesses an array that locates the correct symbol that is indexed by the corresponding hot-spot. Figure 4.5 shows both the array index and the hot-spot declaration associated with the Horizon Line object. It should be noted that the complexity of a particular object may necessitate the declaration of several hot-spots for just one object. In Figure 4.5, the reader can see that there are a total of 35 hot-spots declared for the aforementioned 31 symbols.

```
...
svYhigh[33] =2880
svSymbolPageName[33] = "18. AGL Analog Pointer"

svXlow[34] = 4200        -- AGL Analog Bar
svXhigh[34] = 4380
svYlow[34] = 2280
svYhigh[34] =3705
svSymbolPageName[34] = "17. AGL Analog Bar"

svXlow[35] = 1980     -- Pitch Ladder / Horizon Line
svXhigh[35] = 3345    -- Done Last because of special
svYlow[35] = 2205     -- processing
svYhigh[35] =3070
svSymbolPageName[35] = "4. Pitch Ladder"
...
```

Figure 4.5. Hot-Spot Declaration

The hot-spot declared in Figure 4.5 is not all inclusive for the Horizon Line object. As the comments in the code reveal, the "Pitch Ladder" and Horizon Line occupy nearly the same space, and must have a handler added to determine the exact location of the user's cursor. Since the Horizon Line is surrounded both above and below by the Pitch Ladder (see Figure 4.6), the equation of a line was utilized to determine the user's selection.

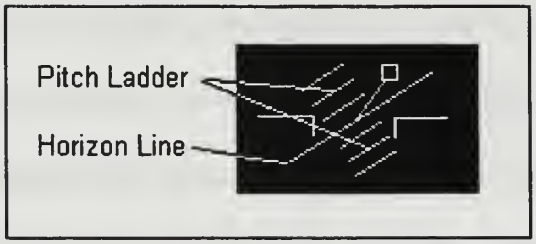

Figure 4.6. Pitch Ladder and Horizon Line

Anything clicked above or below a very thin strip surrounding the Horizon Line but within the boundary declared in Figure 4.5 will bring the user to page 5 of Lesson 1. Figure 4.7 shows the handler for the Horizon Line/Pitch Ladder objects.

```
  if (((xcoord >= svxlow[loopindex]) and (xcoord <=
  svxhigh[loopindex])) and \((ycoord >=
  svylow[loopindex]) and (ycoord <= svyhigh[loopindex])))
        if loopindex = 35
-- use equation of a line to determine if above or below horizon line
-- equation is based on the coordinates for pitch ladder in
-- svylow, svyhigh, svxlow, svxhigh
-- must negate since coordinate system reversed in y direction
                                tempycoord = (-1) * ycoord
                                if tempycoord > ((810/1245) * xcoord) - 4400
                              tempvar = "4. Pitch Ladder"
                            else
                              if tempycoord < ((810/1245) * xcoord) - 4500
                               tempvar = "4. Pitch Ladder"
                              else
                               tempvar = "5. Horizon Line"
                              end
                            end
                          else
                               tempvar  =  svSymbolPageName[loopindex]
                        end
```

Figure 4.7. Handler for Horizon Line/Pitch Ladder hot-spots.

The final excerpt for Module 2, Lesson 1, page 5 is shown in Figure 4.8. This figure shows a portion of the handler that makes the cursor change to the form of a hand whenever the cursor is placed over one of the 31 symbols within the HUD display. Recall from Chapter III that each of the 31 symbols are made up of a composition of *MTB* objects (lines, rectangles, polygons, etc.).

```
if ((( xcoord > 1035) and (xcoord <4485)) and ((ycoord > 1035) and (ycoord < 4485)) and (svPageNum
<> 1))
        if object of target is "rectangle" then
          svoldCursor = sysCursor
          sysCursor = cursor "hand"
  end
        if object of target is "field" then
          svoldCursor = sysCursor
          sysCursor = cursor "hand"
        end if
        if object of target is "angledline" then
          svoldCursor = sysCursor
          sysCursor = cursor "hand"
  end
if object of target is "polygon" then...
```

Figure 4.8. Cursor Handler

In summary, the above example and excerpts pertain to only one page of many. Despite the numerous benefits of object-oriented design, it is not necessarily a means to eliminate the headaches and frustrations often associated with coding and troubleshooting. A significant amount of programming is involved with interactive computer-based training. This point is emphasized in Appendix E, which contains the script for this one page in its entirety.

B. DISCUSSION

The assignment to design or re-engineer an HH-60H ANVIS/HUD computer-based trainer for Commander Naval Air Systems Command was successfully completed within the allotted time-frame.

As is the case with all ISD/SDLC products, however, the life of this CBT is just beginning; hence, there are certain to be follow-on versions of both the HH-60H system

and systems for other rotary-wing aircraft. This section contains several items that should be considered during the field testing, implementation, and support phases of this system.

1. Composition and Limitations

The ANVIS/HUD application is, by CBT standards, designed on sound principles and quality instruction. The trainer effectively avoids the design pitfalls discussed in Chapter II, and it meets all of the requirements stated in the project proposal. Nonetheless, because this was a re-engineering process, there are some structural limitations that may continue to be passed down from generation to generation if a different authoring system is not chosen. While the Introduction pointed out some of the limitations of the system, the following additional points should be considered when assessing the structure of the final product and equipment required to operate it:

A. The final product is biased towards the original version. The navigation interface, text, and graphics were changed only where needed to be in order to convert the system and increase the training benefit. Although virtually hundreds of changes were made, the modular structure of the original CBT remained intact throughout the revision process.

B. The trainer is time consuming. It takes the average user approximately five hours to complete it from start to finish. Despite the trainer's length, however, the efficiency of the CBT is much better than the alternative. When compared to receiving the equivalent amount of data in the classroom and cockpit environments, students will save at least twice as much time by using the trainer.

C. Since the trainer was designed in *MTB* 3.0, test logs, custom controls, and windows-style enhancements that are within the capabilities of *MTB* 4.0 CBT upgrade, are not included in this version.

D. The trainer requires at least 8 MB of RAM to operate effectively. While the CBT will operate on 4 MB of RAM, the navigation and handling is lethargic and extremely delayed - largely due to the high number of graphic files in each lesson. In addition, the minimum processor capability recommended to run the trainer is a Pentium® of no less than 75 MHz. Although specifications from PMA-205 may have called for lower capabilities than these, many of the requested specifications were either slightly unrealistic or the received specifications were outdated. For example, even when the baseline system was tested on 80486SX 25 MHz computers, it took an average of four seconds to navigate between pages - a figure that is unacceptable by CBT standards. Nonetheless, the increase in computer capability should not prove to be a major obstacle. Since most squadrons contain at least one Pentium® PC and several 486 PCs, users should not encounter difficulties finding suitable equipment to house the CBT.

2. Recommendations

Based on the experience gained through this research, the following recommendations are provided for subsequent revisions and future related projects.

With all of the CBT features that are included with the new *MTB* 4.0 edition, many upgrades could be implemented without code manipulation. Follow-on projects should research the possibilities of incorporating test data bank logs, user identification log-ins, and enhanced questioning procedures.

Based on cost, user interviews, and equipment limitations, video and sound enhancements should be incorporated sparingly for the following reasons:

- Users do not necessarily want video and sound in this type of environment. If the trainer is already time-consuming, video and sound may compound that problem.
- As more video and sound files are utilized, more memory will be required to house and operate the system.
- Incorporating these upgrades, especially video, can be expensive. Collecting and digitizing high-quality video footage cannot be accomplished with mediocre equipment.

Future project teams should make every attempt at not only acquiring a set of the ANVIS/HUD devices for the aircraft in question, but also requesting a laboratory demonstration. Hands-on experience with the actual equipment will greatly reduce discrepancies in graphic depiction, documents, and source code.

3. Conclusion

This study of CBT design principles and project re-engineering revealed a process that is complex, extremely powerful, and ever-changing. Computer-based instruction is widely used throughout various industries, and when held to the proper standards, its potential is virtually limitless. In addition, the odds of successfully completing a project are greatly increased by adhering to regimented, methodical, and well-defined developmental techniques within a reasonable agenda.

On a more refined scale, this project produced just what it was intended to: (1) a quality training package on current night-vision equipment for fleet aviators, (2) decreased training costs within the HH-60H community with no reduction in training benefit, (3) a

thorough review of current CBT/MM concepts, standards, success stories, and authoring systems, and (4) a master's level education acquired under real-world time and cost constraints.

Finally, with technological advances on the rise, the availability of multimedia features and upgrades for CBT products will be constantly increasing. Even today, a project designer is tempted to add an egregious number of easy-to-use "bells and whistles" during the ISD/SDLC phases. Future developers should beware, however, that until the user's equipment can accommodate such software advances, the inclusion of multimedia enhancements should be approached with caution. Once placed outside of the academia microcosm, the novice developer quickly realizes, if he has not done so already, that it is the user - not technology - that dictates elaboration.

REFERENCES

Abdel-Hamid, Tarek K., and Stuart E. Madnick, "Lessons Learned from Modeling the Dynamics of Software Development," *Management of Computing*, vol. 32, no. 12, December, 1989.

Alessi, Stephen M., and Stanley R. Trollip, *Computer-based Instruction: Methods and Development*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

Becker, Curtis A., Joy H. Matsumoto, Christopher J. Phillips, and David Kennedy, *Procedure Training for Full-mission Simulation*, Bio-Dynamics Research and Development Corp., Eugene, OR, December, 1990.

Ciavarelli, A. P., *Definition of Learning Objectives for ANVIS/HUD Trainer*, Naval Postgraduate School, Monterey, CA, June, 1996.

Ciavarelli, Anthony, Kishore Sengupta, and Wolfgang Baer, *Night Vision Goggle (NVG) Training Technology Report*, Technology Report, Naval Postgraduate School, Monterey, CA, December, 1994.

Clark, Ruth C., "Authorware, Multimedia, and Instructional Methods," *Macromedia Authorware: Taking the Plunge*, ver. 3, Macromedia, Inc., 1995.

Clemons, Eric K., *Strategic Investments in Information Technology*, Communications of the ACM, Vol. 34, January, 1991.

Epperson, Sean T., *Animation Within a Multimedia Taining System for Night Vision Goggles*, Master's Thesis, Naval Postgraduate School, Monterey, CA, March, 1995.

Gagne, R. M. and L. J. Briggs, *Principles of instructional design*, Holt, Rinehart, and Winston, New York, 1979.

Gery, G., *Making CBT Happen: Prescription for Successful Implementation of Computer-Based Training in Your Organization*, Weingarton Press, Boston, MA, 1987.

Jones, M. K., *Human-Computer Interaction: A Design Guide*, Educational Technology Publications, Englewood Cliffs, NJ, 1988.

Kearsley, Greg, *Computer-Based Training*, Addison-Wesley Publishing Company, Inc., Reading, MA, 1983.

Kearsley, Greg, B. Hunter, and R. J. Seidel, "Two decades of computer-based training," *T.H.E. Journal*, January, February, 1983.

Kemp, J. E., *The Instruction Design Process*, Harper and Row, New York, 1985.

Larkin, J. H. and R. W. Chabay, *Computer-assisted Instruction and Intelligent Tutoring Systems*, Lawrence Earlbaum Associates, Hillsdale, NJ, 1992.

Larsen, William E., Robert J. Randle, Jr., and Lloyd N. Popish, *Vertical Flight Training*, NASA Reference Publication 1373, DOT/FAA/CT-94/83, Moffett Field, CA, 1995.

Lok, Joris J., "Computers Offer Training Efficiency," *Jane's Defence Weekly*, vol. 12, no. 5, August 5, 1989.

Mager, R.F., *Preparing Instructional Objectives*, Lake Publishing, Belmont, CA, 1984.

Marine Aviation Weapons and Tactics Squadron One (MAWTS-1), *Helicopter NVG Manual*, July, 1993.

McLean, Brian W., *Joint Training for Night Air Warfare*, Air University Press, Maxwell Air Force Base, AL, 1992.

Merrill, M. D., "Learner Control in Computer-based Learning," *Computers and Education*, vol. 4, pp. 77-95, 1980.

Merrill, M. D., "Component display theory, in C.M. Reigeluth, (Ed.)," *Instructional Design Theories and Models: An Overview of Their Current Status*, Earlbaum, NJ, 1983.

Montague, W. E., J. A. Willis, and W. H. Wulfeck, II, "Computer-based Instruction: Will It Improve Instructional Quality?", *Training Technology Journal*, vol. 1, no. 2, pp. 4-19, 1983.

Naval Command, Control and Ocean Surveillance Center (NCCOSC), Draft - UH-1N HUD Interactive Courseware Trainer Instructional Design, Research, Development & Evaluation (RDT&E), Division Detachment, Warminister, PA, December 9, 1994.

Oblinger, D., *Introduction to Multimedia Instruction: A Primer*, University of North Carolina Press, 1992.

O'Leary, Les, "Illuminating Unique Military Capabilities," *Jane's Defence Weekly*, vol. 24, no. 23, December 9, 1995.

Ruffner, John W., Gary W. Coker, and Richard D. Weeter, *Development of the AH-64A Display Symbology Training Module*, Research Note, Anacapa Sciences, Fort Rucker, AL, May, 1989.

Troxel, David and Andrew Chappell, "ANVIS/HUD: An Operational and Safety Enhancement for Nap-of-the Earth Night Flight," *U.S. Army Aviation Digest*, U.S. Army Aviation Center, 1-93-2, March/April, 1993.

Whitten, Jeffrey L., Lonnie D. Bentley, and Victor M. Barlow, *Systems Analysis and Design Methods*, Third Edition, Irwin Inc, Boston, 1994.

Williams, Amanda, *The Application of Computer-Assisted Instruction in F/A-18 Pilot Training*, Taining Technology Journal, vol. 1, no. 4, Winter, 1984.

APPPENDIX A. TRAINING REQUIREMENTS ANALYSIS

The following two sections pertain to the training requirements analysis portion of the ISD/SDLC process. Section I is a blank worksheet identifying the questions asked of the subject matter experts during the trip to HCS-5. Section II summarizes the information extracted from the worksheet following the trip.

I. WORKSHEET BLANK:

Initial Points of Contact:

1. CDR Douglas Moret (CO)
2. LCDR Michael Remington (OPS)
3. LCDR Craig Miller (NATOPS)
4. LT William Clatterbuck (Training)

Other Points of Contact

1. _____
2. _____
3. _____
4. _____
5. _____

Squadron Data

1. Phone (Voice) (805) 989-8264
   (DSN) 351- 8264

2. Fax (805) 989-8040

3. Mailing Address:_____
   _____.

4. Comments/Notes: _____
   _____.

DOCUMENTS NEEDED:
1. Current NATOPS for HH-60H
2. Squadron Training Syllabus
3. Training Plan For ANVIS/HUD (if any)
4. Defined Proficiency and Currency Standards

5. Squadron Roster

TRAINING EQUIPMENT AVAILABLE

1. Availability of Trainers and Simulators
   a. Type (OFT, WST, Procedures, etc.)
   b. Brief Description (include manufacturer)_____
   _____.
   c. Location of Simulation Equipment _____.
   d. Primary Mission/Tasks Trained on Simulator _____.

2. Availability of CBT, and/or PC Equipment
   a. Type Computing Equipment (IBM Compatible, Model and Series) _____.
   _____.
   b. Type Software Available (Operating System, and Applications) _____
   _____.
   c. Any PC Training Software available now? No [ ] Yes [ ], Type_____
   and current applications, and expected upgrades -_____.

MISSION ANALYSIS

Primary Mission: Please describe the role of the HH-60H in the Navy, and your primary mission responsibility.

_____
_____
_____

1. Scenario: Can you provide us with a narrative description of a "typical" operational scenario for your Primary Mission?

Brief:
Preflight:
Takeoff:
Enroute Profile:
Mission Tasks:
Return Profile:
Landing:
Postflight Shutdown:
Debrief:

2. Secondary Mission: Can you provide us with a brief narrative scenario (as above) of of another type of operational mission?

INTERVIEW QUESTIONS

1. What percentage of your operational missions are flown at night?

2. Has there been increased emphasis on night flying operations?

3. Describe any problems and shortcomings that you may have experienced while conducting night flight operations?

4. What specialized education and training do aviators get on night operations and/or night doctrine and systems?

5. Tell us about your night qualifications program (training standards, syllabus, and flight qualifications program)?

6. Are there any plans/thoughts underway concerning the introduction of the ANVIS/HUD System?

7. Are there any special circumstances, conditions, expected problems, etc. regarding the introduction of the ANVIS/HUD?

8. Do you have any experience using the ANVIS/HUD prototype, and if so can you tell us about your experience, impressions, and thoughts on effective use of this system?

9. What kinds of problems, if any, do you anticipate regarding the introduction and use of the ANVIS/HUD in performance of your mission?

10. Do you have any thoughts/ideas about what may be important to include in a training program for ANVIS/HUD?

11. Do you have any suggestions concerning how such a training system will work best in your unit?

12. We would like to come back for a formal photo shoot, can you tell us what would be required to make this possible?

13. Would it be possible to take a "walking tour" through your facility, and aircraft cockpit ?

14. Are there any objections if we take some preliminary photo/video footage of selected facility locations and/or aircraft? This will help us prepare for a more formal photo session to be arranged at a later date.

II. TRAINING REQUIREMENTS ANALYSIS RESULTS

A. The HCS Community

1. Overview

An HCS squadron is comprised up of reservists, and most existing reserve squadrons (RESFORONs) mirror fleet squadrons in day-to-day operations. Squadrons are made up of TAR (Training and Administration of Reserves) personnel and SELRES (Selected Reserves) personnel. TARs provide the stability in the squadron - working full work weeks, expediting administrative matters, coordinating deployments, and implementing training plans. SELRESs are customarily the "weekend warrior" type, serving in squadrons in order to supplement their income, earn flight hours, and remain members of an elite military team that defends U.S. interests abroad. Several SELRES personnel actually make HCS a full-time occupation, but most do not.

2. Primary Missions

- Combat SAR (Strike Rescue). Crew composition: 2 pilots, 3-4 aircrewman (2 gunners, one corpsman, one standby). Crews extract downed pilots and stranded military personnel from hostile/combat/conflict ridden areas. Aircraft can "officially" hold 4 survivors. 250 NM range. Flight profiles (brief topics, preflight, etc.) can vary immensely.
- Special Warfare (SPECWAR). Crew Composition: 2 pilots, 1-3 aircrewman. Crews assist SPECWAR teams (SEALS, RANGERS, etc.) on insertion/extraction missions in potentially hostile LZs, littoral areas, and open-ocean drops. Aircraft can hold 8 troops (SPECWAR team members). 200 NM range. Again, flight profiles can vary immensely.

3. Deployment

HCS squadrons prepare selected teams - or detachments - for action, and squadrons always keep at least one detachment "at the ready" for immediate deployment. Typical deployment configurations are usually anything but typical. HCS squadrons support such a wide variety of missions, that a "standardized" detachment is virtually impossible to maintain since every mission calls for a slightly different configuration. Nonetheless, HCS planners use a 40-man detachment concept as a guideline for meeting possible commitments. The 40-man concept supports 2 aircraft: 8-9 pilots, 8-9 aircrew, 22-24 maintenance and support personnel. Detachments can:

- operate independently at sea (on carriers or small boys)
- operate independently ashore (bringing with them requisite personnel to

perform a shore-based mission (mess specialists, corpsmen, etc.)
- augment HS squadrons

B. Initial Points of Contact

- CDR Ray Bellant (XO)
- LCDR Mike Remington (OPS)
- LCDR Craig Miller (TACTICS)
- LCDR Matt Ragan (ADMIN)
- LT Bill Clatterbuck (TRNG)
- LT Pat Davitt (CBT Coordinator) - pjdavitt@aol.com
- LCDR Joe Vaughn (Former HCS-5/TPS Pilot - currently in NPGS OA curriculum
  Home phone: (408) 655-5056
- Pt. Mugu photo lab: DSN 351-7824.

Comments: Bill Clatterbuck and Craig Miller were exceptionally helpful. Both have extensive NVG experience, and Bill saw the initial versions of the ANVIS/HUD while stationed at HCS-4.

C. Documents/Material Acquired or Incoming

- LSI Aircrew Trainee Guide for HCS-4 and HCS-5
- Personnel Qualification Standard (PQS) for HH-60H pilot positions
- HCS-5 Pilot and Aircrew Roster
- HCS-5 Standard Operating Procedures (SOP) Instruction 3710.1M; 6 MAY 1995
- COMHELWINGRES NVG Use and Training Doctrine Instruction 3500.8F
- One cover photo. Have the number to the photo lab. Several more photos are enroute from Pat Davitt and Matt Ragan. (Pat Davitt has flat-bed scanner and will digitize and e-mail us graphics. See e-mail address above.

D. Training Breakdown

1. Overview

Initial training is conducted at the FRS (in either San Diego or Mayport). Pilots only learn non-mission specific procedures (EPs, etc.) and then they continue on to their respective assignments. It is at the HCS squadrons where they receive OJT on the mission-related procedures. Mission-related WSTs, WTTs, or stationery simulators do not exist. However, the west-coast community does have access to a NITE lab down in San Diego, but training is not mandatory.

As mentioned in Section C above, we do have a copy of the HCS-5 training doctrine which outlines - in detail - required learning objectives and theory utilized in NVG training missions

2. Equipment/Location

- Over 100 PC's, 15 486s, 10 (and growing) Pentiums®.
- 4 WICAT stations. Supported by a commercial contractor.

Comment: Interviews suggested that once the ANVIS/HUD CBT arrives, it will be installed, at a minimum, on a Pentium® machine and located in the WICAT/Training room just next to the Training Department's office. Readily accessible, it seats up to 10 people. There is no shortage of hardware, and the squadron is allocated even more Pentiums® for FY'96. The squadron anticipates ultimately loading the ANVIS/HUD package onto numerous computers throughout the squadron.

There is currently no structured plan for ANVIS/HUD training/CBT utilization, but the perception is that there will be one by the July time-frame.

E. Interview Excerpts

1. On CBTs:

- Rectangular box on new ANVIS HUD display is not visible.
- Full display is not visible w/ the 18mm tubes, he expects the 24mm tubes to fix that.
- Didn't like touch-screen CBTs. Inaccurate and inconsistent. "With mouse driven CBTs, you always no where to push, and you can pin-point items easier."
- Didn't like WICAT/Network CBTs. "WICATs crash all the time - your (ANVIS/HUD) system is portable, low maintenance...that's a lot better."
- "Don't add video and animation just for the sake of making it fancy. If it helps get the point across, fine. But if some video is run 100 times for 15 seconds at a wack, and every time it runs I'm just sitting around waiting for it to end, that's 1500 seconds of my time that I'd rather be spending on something else. As you know...time is money."

2. On NVG Operatons:

- Likes the idea of a HUD because it will reduce the scan load. The RADALT and the MASTER CAUTION pack (the big 5) will always be displayed on the HUD and pilots won't have to play games with the goggles to see the minimum instruments. With the 25 mm lenses, scanning the RADALT while sitting in the right seat requires the pilot to turn his head to the right and eyes to the left. Very disorienting...takes a while to get used to. Even with 15mm's, scanning is a problem. Some pilots place the goggles a short distance away from their faces so that they can "peek" under the goggles

at the flight instruments - thereby reducing the display diameter by up to 20 or 30 percent.

- Crewman is often the first person to see the MASTER CAUTION, "since the copilot is running 'God's Eye' Navigation, FLIR and NVGs...the MASTER CAUTION panel is often the last thing that gets brought into the scan."
- Has heard rumors that the roll rate, air speed, and RADALT displays on the new HUD system are delayed (2, 1, and 1 second respectively).
- Concerned about the location of the ALE-39 and ANVIS/HUD collective switches. Both are controlled by the left thumb.
- Squadron flies about 30% night operations (OPS Officer verified). Night flight time is limited to the field's operating hours (it closes at 2300). Squadron has an arrangement with base ops to conduct close-field operations, but it rarely exercises that clause for training missions.

F. Conclusion

Squadron liaison successful. Upper management shows strong support for follow-on interviews and photo/video collection trips.

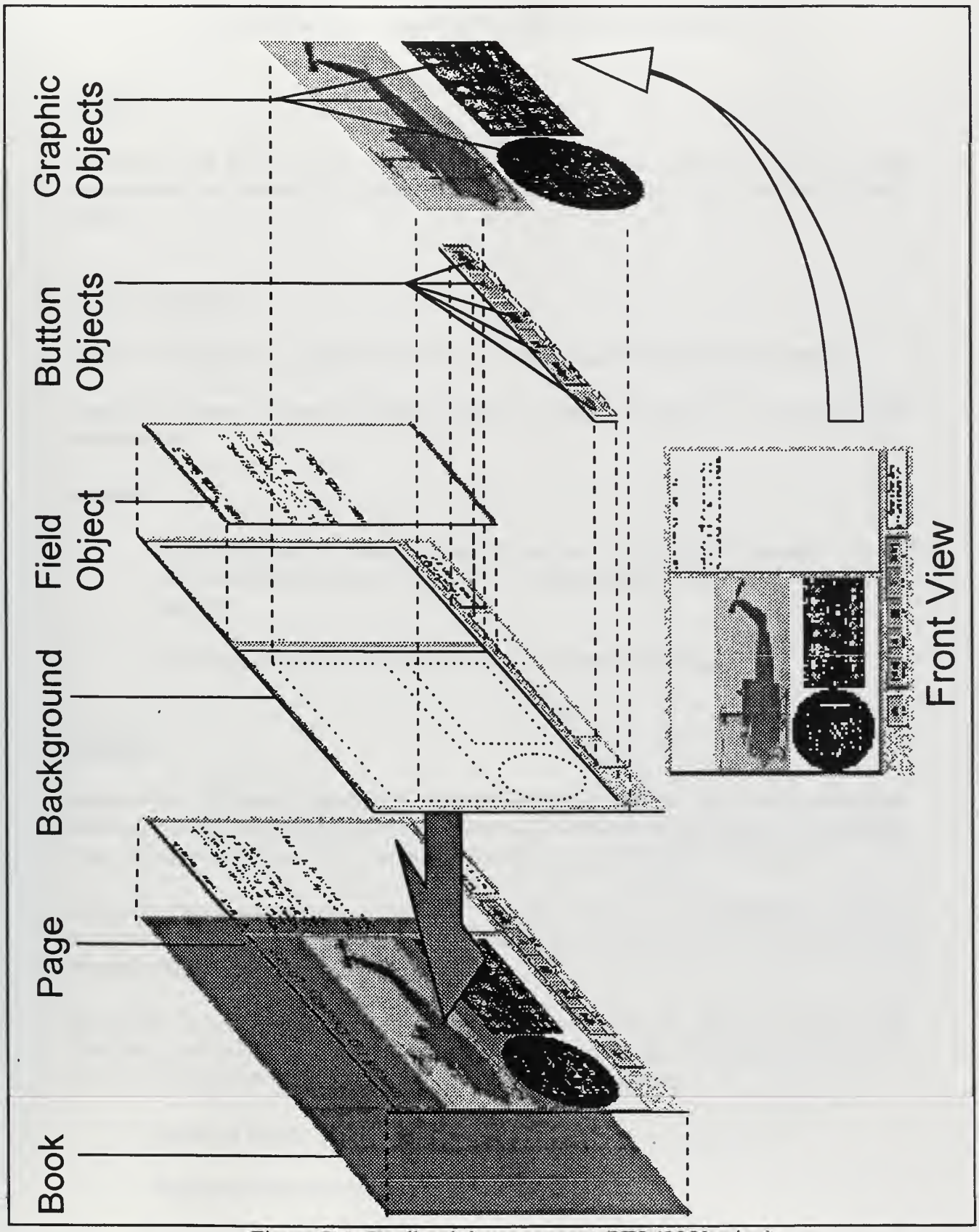

Figure B.1. Toolbook layer concept (UH-1N Version)

APPENDIX C. CHANGES AND ENHANCEMENTS

Global:

Throughout the entire trainer, all UH-1N helicopter-related graphics, text, and coding were replaced as necessary to ensure the final product accurately represented an HH-60H system.

Overview Module:

General Description: Comprised of four mini-books (sections), totaling 55 pages.

General Changes: Changes included graphic and text modification, and limited code modifications.

Specifics:

- On warnings and cautions, replaced all red text with black, and changed background from baby-blue to red for warnings, and from green to yellow for cautions.

- Modified several text fields to improve on awkward wording.

Module 1:

General Description: Comprised of four mini-books (3 lessons, 1 test), with each lesson containing at least one imbedded exercise (lesson 2 contains two exercises). Exercises and tests are generated from a question bank through a random function. 125 pages.

General Changes: Changes included graphic and text modification, program modifications (approximately 300 lines of code (LOC) were modified or affected), and original graphic generation.

Specifics: In addition to changes made that were similar to those discussed in the Overview, the following graphics were either developed in MS Paintbrush and refined in Aldus Photostyler or modified in the *MTB* bitmap-edit application:

- Replaced Master Display symbology

- Replaced Generator Mode Display

- Replaced collective graphics

- Replaced circuit-breaker panel graphics

- Replaced SDC component graphics

- Rewrote all appropriate test and exercise questions

Module 2:

General Description: Code intensive module, with a high level of interaction between the system and user. Comprised of four mini-books (three lessons, one test), with each lesson containing one imbedded exercise, and the test containing six more imbedded mini-books. Exercises and tests are generated from a question bank through a random function. 79 pages.

General Changes: Changes included graphic and text modification, original graphic generation, and re-engineering of over 3000 LOC.

Specifics:

- Replaced every symbol of the display unit with the correct HH-60H symbol, and assigned each symbol a new identification number and OOD layer. Figure D.1 shows the difference in symbology between the two systems.

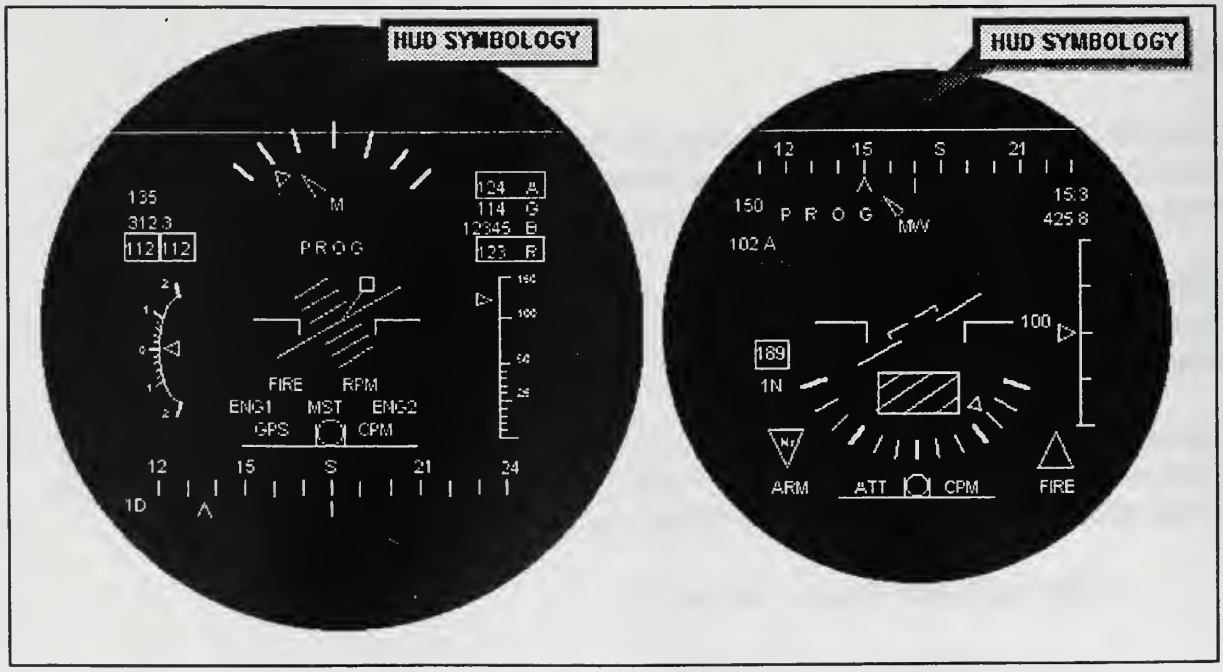

Figure C.1 Display Units (HH-60H on left, UH-1N on right)

- Reconfigured all arrays, pointers and loops to reflect the proper number of symbols.

- In accordance with HH-60H procedures, reworked each lesson, exercise and test to ensure symbols flashed in the correct sequence.

- Slowed the movement of graphic properties to allow the student to better understand the lesson's intent. By slowing the graphics' movements, the student can now clearly see the results of his interactions (pp. 8 - 10).

- Reconfigured the test and exercise questions during the programming phase. Symbols to be selected are listed in the order in which they flash on the display to allow the student to better concentrate on the task of programming. Original version listed the symbols randomly, causing the student to waste time scanning the list vice scanning the display.

- Incorporated a handler to remind student to program in the correct mode during the test. Without the handler, student was often confused during the review portion when he didn't see his actions reflected in the display labeled "Your Answer".

Module 3:

General Description: Comprised of four mini-books (three lessons, one test), with each lesson containing one imbedded exercise. 83 pages.

General Changes: Changes included graphic and text modification, original graphic generation, and re-engineering of less than 200 LOC. Limited inter-actively.

Specifics: Significant changes included numbering egress procedures, modifying caution and warning colors, and converting "Ok" buttons to "OK" buttons.

Module 4:

General Description: Comprised of seven mini-books (six lessons, one test), with five of six lessons containing one imbedded exercise. Lesson six is highly interactive. 97 pages.

General Changes: Changes to the lessons included graphic, text modification, and programming modifications (approximately 500 LOC).

Specifics:

- Graphic modifications were made to pages in order to assist the user to locate specific HUD-related components (Converter Control Unit (CCU), Signal Data Converter (SDC), etc.).

- Created graphic objects that improved the display unit representations for the troubleshooting portion of the exercise (i.e. created a group object to represent an out-of-focus display unit.

APPENDIX D. REFINEMENTS

The following includes descriptions of those areas throughout the original system that required corrections or refinements

Module 1

- M1L2: Fixed improper navigation when the "GO TO..." function was utilized and page 12 was selected. Instead of going to p. 12 of 25, the CBT took the student to p. 12 of 30. Pp. (12 of 30 - 16 of 30) were embedded, and they were supposed to remain hidden. Reconfigured "GO TO..." handler to function properly.

- M1L2; p. 13 (SDC Exercise):

  - Fixed exercise title to reflect proper number of available questions and removed global variable name from title (i.e. "Question 2 of 7" vice "Question 2 of svQTotal").

  - Fixed "GO TO..." function. "GO TO" was inoperative throughout the exercise.

  - Repositioned graphics throughout exercise to reveal masked questions and associated text.

- M1L2; p. 19b: Repaginated p. 19b of 30 to 19b of 25 (script)

- M1L2; p. 25 (CCU Exercise): Fixed inoperative rewind button.

- M1L3: Reconfigured "save on close" option to ensure user was *not* prompted to save the module on exit.

Module 2

- M2L1: Reconfigured "save on close" option to ensure user was *not* prompted to save the module on exit.

- M2L1: Repositioned "Exit" prompt to ensure "M2L1.Text" did not mask the "Exit" function.

- M2L2; p. 13: Replaced display units that contained *inactive* field objects (user cannot modify) vice *active* field objects (user can modify).

- M2L2; p. 15: Fixed inoperative "Fast-Forward" button.

- M2L2: Reconfigured background code to ensure user did not receive an error message when navigating from any other page in the lesson back to p. 1. Originally, the user received an error, *"There is no object 'DisplayId'"* and had to hit escape to continue.

- M2L2; p. 16: Reconfigured code to ensure user does not see the message "*Loading Lesson 3*" until he chooses to navigate to Lesson 3. Original version erroneously showed that message as soon as the user navigated to the introductory page of the exercise.

- M2L2; p. 17 (Programming Exercise): Reconfigured the programming switch graphic object to be hidden whenever an error message appeared over the CCU panel.

- M2L2: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M2L3: Reset system settings to ensure the user is not prompted to save on exit.

- M2L3: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M2L3: Reconfigured background code to ensure user did not receive an error message when navigating from any other page in the lesson back to p. 1. Originally, the user received an error, *"There is no object 'Playbuttondown'"* and had to hit escape to continue.

- M2L3; p. 15 (Pitch and Roll Exercise):

    - Reconfigured code to alert the user with an appropriate error message when the user was performing an improper procedure. Original version did not alert the user until it was too late, deeming his actions irreversible. This forced the user to abandon the final question of the test.

    - Reconfigured code to ensure proper handling of dialog boxes. Original version did not "*hide CCU panel*" in order to "*show Error Message*".

- M2 PRACTICE: Created code to display error messages and handlers in the event the user tried to use any of the buttons other than "Exit" to navigate around the practice lessons.

- M2TEST; Questions 8-9:

  - Reconfigured the programming switch graphic object to be hidden whenever an error message appeared over the CCU panel.

  - Removed endless loop from coding to alert user that Question 9 was complete and allow navigation to Question 10.

  - Fixed inoperative "Rewind" button.

- M2TEST; Review Question 9: Completely restructured code for the test and review "exe" files to correct various errors culminating in the proper review of Question 9. Original version performed several incorrect functions including:

  - Did not show declutter symbols in "Correct Answer" display.

  - Showed numerous incorrect symbols on "Correct Answer" display.

  - Did not allow the user to receive credit for a correct answer.

Module 3

- M3L1: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M3L2: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M3L2; p.4: Rewrote code and originated graphics to ensure the user did not receive an error message stating "*No object named 'FailLamp'*", and directed the user's attention to the disappearing "*FailLamp*" light.

- M3L3: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

Module 4

- M4 (Menu): Reconfigured the SDC, CCU, & DU buttons to ensure that all 3 sent user to the appropriate page in the lesson. Original version sent the user to p. 32 of 38 regardless of the button selected.

- M4L1: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M4L1; p. 5: Fixed the "*Next Lesson*" button so that the button does not display the right half of a red dialog box when transitioning to the next lesson.

- M4L2: Reconfigured "Exit" code to ensure "GO TO..." viewer was hidden if "Exit" was selected. If the "GO TO..." viewer was visible in the original version and "Exit" was selected, the "Exit" confirmation button was masked by the "GO TO..." viewer.

- M4L2 (Exercise):

    - Question 5: Typo: changed "lefe" to "life"

    - Question 9: Reworded question. The correct answer with the original wording was wrong.

- M4L3; p. 17 (Exercise):

    - Fixed exercise title to reflect proper number of available questions and removed global variable name from title (i.e. "Question 2 of 7" vice "Question 2 of svQTotal").

    - Fixed Questions 1, 2, and 3. Original version displayed an incorrect answer when prompted to display the correct answer.

- M4L3; p. 21a: Fixed text to reflect proper page numbers (changed "a through e" to "b through e")

- M4L6; p. 7 (Troubleshooting Exercise):

    - Enabled user to exit the lesson during exercise. Original version masked the exit confirmation box with any one of 5 viewers. New version hides viewers and reveals the exit confirmation box. If user chooses to cancel, he can start over on the current question.

- Enabled the "Fast-Forward" and "Rewind" buttons.

- Fixed the "Play" button to show a depressed configuration when clicked.

APPENDIX E. SCRIPT FOR MODULE 2, LESSON 1, PAGE 5

```
-- Script for "Mod2Les1"

to handle enterApplication

      system INT svPagenum                    -- will contain the page
                                                 number of the display
                                                 text viewer
      system INT svPrevPageNum                -- will contain the
                                                 previous page number
                                                 of the display text
                                                 viewer
      system INT svXlow [35]                  -- coordinate pairs to
                                                 determine if
      system INT svXhigh [35]                 -- button clicked within
                                                 area
      system INT svYlow [35]
      system INT svYHigh [35]
      system INT svLineStartX [36]            --  X coordinate for
                                                  line start of symbol
      system INT svLineStartY [36]            --  Y coordinate for
                                                  line start of symbol
      system INT svLineEndX[36]               --  X coordinate for
                                                  line end (at viewer)
      system INT svLineEndY[36]               --  Y coordinate for
                                                  line end (at viewer)
      system svSymbolPageName[35]             --  Page Names for
                                                  button clicks
      system svNameSymbol[30]                 -- for exercises
      system INT SymbolIndex[30]              -- index to arrow table
      system svOldCursor

      svPagenum = 1                           -- start on Page 1
      svPrevPageNum = 1

      set text of field "Pagenum" to "1 of 33"

 -- initialize the name array for the exercises

      svNameSymbol[1] = "Aircraft Fixed Reference"
      SymbolIndex[1] = 3

      svNameSymbol[2] = "Pitch Ladder"
      SymbolIndex[2] = 4

      svNameSymbol[3] = "Horizon Line"
      SymbolIndex[3] = 6

      svNameSymbol[4] = "Angle of Roll Scale"
      SymbolIndex[4] = 7

      svNameSymbol[5] = "Angle of Roll Pointer"
      SymbolIndex[5] = 8

      svNameSymbol[6] = "Left and Right Engine Torque"
      SymbolIndex[6] = 9
```

```
svNameSymbol[7] = "Torque Warning Box"
SymbolIndex[7] = 10

svNameSymbol[8] = "Barometric Altitude Digital"
SymbolIndex[8] = 11

svNameSymbol[9] = "Aircraft Heading Fixed Index"
SymbolIndex[9] = 12

svNameSymbol[10] = "Compass Reference Scale"
SymbolIndex[10] = 13

svNameSymbol[11] = "Bearing to Waypoint Analog"
SymbolIndex[11] = 14

svNameSymbol[12] = "Heading to Waypoint Digital"
SymbolIndex[12] = 15

svNameSymbol[13] = "Ground Speed Digital (KTS)"
SymbolIndex[13] = 16

svNameSymbol[14] = "Low Altitude Warning Box"
SymbolIndex[14] = 17

svNameSymbol[15] = "AGL Analog Bar"
SymbolIndex[15] = 18

svNameSymbol[16] = "AGL Analog Pointer"
SymbolIndex[16] = 19

svNameSymbol[17] = "RadAlt Digital"
SymbolIndex[17] = 20

svNameSymbol[18] = "True Airspeed"
SymbolIndex[18] = 21

svNameSymbol[19] = "Range to Waypoint, Digital"
SymbolIndex[19] = 22

svNameSymbol[20] = "Master Caution Warning"
SymbolIndex[20] = 23

svNameSymbol[21] = "Trim (Slide Ball)"
SymbolIndex[21] = 24

svNameSymbol[22] = "HUD System Fail Warning"
SymbolIndex[22] = 25

svNameSymbol[23] = "Navigation System Fail"
SymbolIndex[23] = 29

svNameSymbol[24] = "Velocity Vector"
SymbolIndex[24] = 30

svNameSymbol[25] = "Airspeed Warning Box"
SymbolIndex[25] = 31
```

```
    svNameSymbol[26] = "Vertical Speed Scale"
    SymbolIndex[26] = 32

    svNameSymbol[27] = "Vertical Speed Scale Pointer"
    SymbolIndex[27] = 33

    svNameSymbol[28] = "Quadrant Threat Warning"
    SymbolIndex[28] = 34

    svNameSymbol[29] = "PROG/ADJ Indicators"
    SymbolIndex[29] = 35

    svNameSymbol[30] = "Mode Number/Declutter Marker"
    SymbolIndex[30] = 36


-- initialize the page name array

-- initialize the jump coordinate table

   svXlow[1] = 1155          -- Range 1
   svXhigh[1] = 1425
   svYlow[1] = 2220
   svYhigh[1] =2730
   svSymbolPageName[1] = "28. Vertical Speed Scale"

   svXlow[2] = 1155          -- Range 2
   svXhigh[2] = 1410
   svYlow[2] = 2895
   svYhigh[2] =3450
   svSymbolPageName[2] = "28. Vertical Speed Scale"

   svXlow[3] = 3780          -- Barometric Altitude, Digital
   svXhigh[3] = 4515
   svYlow[3] = 1770
   svYhigh[3] =1920
   svSymbolPageName[3] = "10. Barometric Altitude Digital"

   svXlow[4] = 1065          -- Heading to Waypoint Digital
   svXhigh[4] = 1410
   svYlow[4] = 1425
   svYhigh[4] =1575
   svSymbolPageName[4] = "14. Heading to Waypoint Digital"

   svXlow[5] = 1110          -- Compass Reference Scale
   svXhigh[5] = 4320
   svYlow[5] = 3915
   svYhigh[5] =4200
   svSymbolPageName[5] = "12. Compass Reference Scale"

   svXlow[6] = 1065          -- Range to Waypoint, Digital
   svXhigh[6] = 1500
   svYlow[6] = 1620
   svYhigh[6] =1815
   svSymbolPageName[6] = "21. Range to Waypoint, Digital"

   svXlow[7] = 1125          -- Left and Right Engine Torque
   svXhigh[7] = 1545
```

```
svYlow[7] = 1920
svYhigh[7] =2070
svSymbolPageName[7] = "8. Left and Right Engine TorqueT"

svXlow[8] = 1065
svXHigh[8] = 1710
svYlow[8] = 1845
svYhigh[8] = 2115
svSymbolPageName[8] = "9. Torque Warning Box"

svXlow[9] = 975          -- Mode Number
svXhigh[9] = 1200
svYlow[9] = 4200
svYhigh[9] =4350
svSymbolPageName[9] = "32. Mode Number/Declutter Marker"

svXlow[10] = 3000        -- Velocity Vector
svXhigh[10] = 3180
svYlow[10] = 2205
svYhigh[10] = 2340
svSymbolPageName[10] = "26. Velocity Vector"

svXlow[11] = 1290        -- Vertical Speed Scale Pointer
svXhigh[11] = 1425
svYlow[11] = 2745
svYhigh[11] =2895
svSymbolPageName[11] = "29. Vertical Speed Scale Pointer"

svXlow[12] = 10          -- Prog / ADJ
svXhigh[12] = 20
svYlow[12] = 10
svYhigh[12] =20
svSymbolPageName[12] = "31. PROG/ADJ Indicators"

svXlow[13] = 3105        -- Angle of Roll Scale  range 2
svXhigh[13] = 3660
svYlow[13] = 945
svYhigh[13] =1470
svSymbolPageName[13] = "6. Angle of Roll Scale"

svXlow[14] = 2160        -- Aircraft Fixed Reference
svXhigh[14] = 2550
svYlow[14] = 2560
svYhigh[14] =2745
svSymbolPageName[14] = "3. Aircraft Fixed Reference"

svXlow[15] = 1955        -- Angle of Roll Scale  range 1
svXhigh[15] = 2290
svYlow[15] = 945
svYhigh[15] = 1425
svSymbolPageName[15] = "6. Angle of Roll Scale"

svXlow[16] = 2220        -- Bearing to Waypoint Analog
svXhigh[16] = 2400
svYlow[16] = 4210
svYhigh[16] =4410
svSymbolPageName[16] = "13. Bearing To Waypoint Analog"
```

```
svXlow[17] = 2010          -- Trim - Range 1
svXhigh[17] = 3540
svYlow[17] = 3750
svYhigh[17] =3780
svSymbolPageName[17] = "23. Trim (Slide Ball)"

svXlow[18] = 2115          -- Navigation System Fail Message
svXhigh[18] = 2475
svYlow[18] = 3570
svYhigh[18] =3730
svSymbolPageName[18] = "25. Navigation System Fail"

svXlow[19] = 1845          -- Master Caution Warning
svXhigh[19] = 3615
svYlow[19] = 3165
svYhigh[19] =3525
svSymbolPageName[19] = "22. Master Caution Warning"

svXlow[20] = 2565          -- Quadrant Threat Warning
svXhigh[20] = 3075
svYlow[20] = 1380
svYhigh[20] =1905
svSymbolPageName[20] = "30. Quadrant Threat Warning"

svXlow[21] = 2610          -- Trim - Range 2
svXhigh[21] = 2880
svYlow[21] = 3525
svYhigh[21] =3750
svSymbolPageName[21] = "23. Trim (Slide Ball)"

svXlow[22] = 2630          -- Aircraft Heading Fixed Index
svXhigh[22] = 2800
svYlow[22] = 4210
svYhigh[22] =4400
svSymbolPageName[22] = "11. Aircraft Heading Fixed Index"

svXlow[23] = 3060          -- HUD System Fail Message
svXhigh[23] = 3405
svYlow[23] = 3570
svYhigh[23] =3730
svSymbolPageName[23] = "24. HUD System Fail Warning"

svXlow[24] = 3120          -- Aircraft Fixed Reference
svXhigh[24] = 3525
svYlow[24] = 2580
svYhigh[24] =2760
svSymbolPageName[24] = "3. Aircraft Fixed Reference"

svXlow[25] = 2535        -- PROG / OK FAIL
svXhigh[25] = 3105
svYlow[25] = 1935
svYhigh[25] =2055
svSymbolPageName[25] = "31. Prog/Adj Indicators"

svXlow[26] = 2295          -- Angle of Roll Pointer
svXhigh[26] = 2460
svYlow[26] = 1305
svYhigh[26] =1500
```

```
svSymbolPageName[26] = "7. Angle of Roll Pointer"

svXlow[27] = 2400          -- Angle of Roll Scale  range 3
svXhigh[27] = 3195
svYlow[27] = 855
svYhigh[27] =1290
svSymbolPageName[27] = "6. Angle of Roll Scale"

svXlow[28] = 3960          -- RadAlt Digital
svXhigh[28] = 4555
svYlow[28] = 2040
svYhigh[28] =2145
svSymbolPageName[28] = "19. RadAlt Digital"

svXlow[29] = 3945          -- True Airspeed
svXhigh[29] = 4500
svYlow[29] = 1410
svYhigh[29] =1530
svSymbolPageName[29] = "20. True Airspeed"

svXlow[30] = 3885          -- Airspeed Warning Box
svXhigh[30] = 4575
svYlow[30] = 1305
svYhigh[30] =1590
svSymbolPageName[30] = "27. Airspeed Warning Box"

svXlow[31] = 3900          -- Ground Speed Digital (KTS)
svXhigh[31] = 4500
svYlow[31] = 1620
svYhigh[31] =1755
svSymbolPageName[31] = "15. Ground Speed Digital (KTS)"

svXlow[32] = 3945          -- Low Altitude Warning Box
svXhigh[32] = 4560
svYlow[32] = 1925
svYhigh[32] =2205
svSymbolPageName[32] = "16. Low Altitude Warning Box"

svXlow[33] = 4035          -- AGL Pointer
svXhigh[33] = 4195
svYlow[33] = 2760
svYhigh[33] =2880
svSymbolPageName[33] = "18. AGL Analog Pointer"

svXlow[34] = 4200          -- AGL Analog Bar
svXhigh[34] = 4380
svYlow[34] = 2280
svYhigh[34] =3705
svSymbolPageName[34] = "17. AGL Analog Bar"

svXlow[35] = 1980          -- Pitch Ladder / Horizon Line
svXhigh[35] = 3345         -- Done Last because of special processing
svYlow[35] = 2205
svYhigh[35] =3070
svSymbolPageName[35] = "4. Pitch Ladder"
```

```
fill svLineEndX with 5580                -- all lines end in
                                            same place for now
fill svLineEndY with 4250

svLineEndX[23] = 5580                    -- special case for AGL
                                            Analog bar
svLineEndY[23] = 3950

svLineStartX[1] = 0                      -- Synopsis display
                                            no line
svLineStartY[1] = 0

svLineStartX[2] = 0                      -- Master Mode
                                            display - no line
svLineStartY[2] = 0

svLineStartX[3] = 2355                   --   Aircraft Fixed
                                              Reference
svLineStartY[3] = 2580
svLineEndX[3] = 2070
svLineEndY[3] = 2025

svLineStartX[4] = 2685                   --   Pitch Ladder
svLineStartY[4] = 2440
svLineEndX[4] = 1995
svLineEndY[4] = 2220

svLineStartX[5] = 2685                   --   Pitch Ladder
svLineStartY[5] = 2440
svLineEndX[5] = 1995
svLineEndY[5] = 2220

svLineStartX[6] = 3255                   --   Horizon Line
svLineStartY[6] = 2340
svLineEndX[6] = 3720
svLineEndY[6] = 2460

svLineStartX[7] = 3270                   --   Ang of Roll Scale
svLineStartY[7] = 1290
svLineEndX[7] = 3630
svLineEndY[7] = 1770

svLineStartX[8] = 2370                   --   Ang of Roll Pointr
svLineStartY[8] = 1425
svLineEndX[8] = 2130
svLineEndY[8] = 1935

svLineStartX[9] = 1230                   --   Lft & Rt Eng Tq
svLineStartY[9] = 1995
svLineEndX[9] = 675
svLineEndY[9] = 2475

svLineStartX[10] = 1245                  --   Torque Warning Box
svLineStartY[10] = 2115
svLineEndX[10] = 870
svLineEndY[10] = 2415

svLineStartX[11] = 3900                  --   Bar Alt, Digital
```

```
svLineStartY[11] = 1845
svLineEndX[11] = 3375
svLineEndY[11] = 1935

svLineStartX[12] = 2730                    --   A/C Fixd HDG Indx
svLineStartY[12] = 4335
svLineEndX[12] = 3120
svLineEndY[12] = 4710

svLineStartX[13] = 2940                    --    Compass Ref Scale
svLineStartY[13] = 4140
svLineEndX[13] = 3105
svLineEndY[13] = 4665

svLineStartX[14] = 2280                    --    Bearing to
                                                 Waypoint Analog
                                                 (normal direction)
svLineStartY[14] = 4350
svLineEndX[14] = 1785
svLineEndY[14] = 4665

svLineStartX[15] = 1110                    -- Heading to
                                              Waypoint Digital
svLineStartY[15] = 1500
svLineEndX[15] = 675
svLineEndY[15] = 1890

svLineStartX[16] = 4400                    -- Grd Spd Digital (KTS)
svLineStartY[16] = 1695
svLineEndX[16] = 4875
svLineEndY[16] = 2040

svLineStartX[17] = 4385                    -- Low Alt Warning (Box)
svLineStartY[17] = 2145
svLineEndX[17] = 5010
svLineEndY[17] = 2355

svLineStartX[18] = 4260                    -- AGL Analog Bar
svLineStartY[18] = 3075
svLineEndX[18] = 4860
svLineEndY[18] = 3475

svLineStartX[19] = 4050                    -- AGL Pointer
svLineStartY[19] = 2850
svLineEndX[19] = 3510
svLineEndY[19] = 3075

svLineStartX[20] = 3990                    -- Rad Alt Digital
svLineStartY[20] = 2085
svLineEndX[20] = 3795
svLineEndY[20] = 2520

svLineStartX[21] = 4065                    -- True Airspeed
svLineStartY[21] = 1455
svLineEndX[21] = 3765
svLineEndY[21] = 960

svLineStartX[22] = 1470                    -- Low RPM Warning
```

```
svLineStartY[22] = 1740
svLineEndX[22] = 2145
svLineEndY[22] = 1695

svLineStartX[23] = 3075                    -- Master Caution Warn'g
svLineStartY[23] = 3165
svLineEndX[23] = 3465
svLineEndY[23] = 2935

svLineStartX[24] = 2730                    -- Trim (Slide Ball)
svLineStartY[24] = 3675
svLineEndX[24] = 2565
svLineEndY[24] = 4155

svLineStartX[25] = 3285                    -- HUD Fail Message
svLineStartY[25] = 3690
svLineEndX[25] = 3345
svLineEndY[25] = 4275


svLineStartX[26] = 3285                    -- NAV Sys Fail Warning
svLineStartY[26] = 3690
svLineEndX[26] = 3345
svLineEndY[26] = 4275

svLineStartX[27] = 3285                    -- PGM Message
svLineStartY[27] = 3690
svLineEndX[27] = 3345
svLineEndY[27] = 4275

svLineStartX[28] = 3285                    -- NAV Message
svLineStartY[28] = 3690
svLineEndX[28] = 3345
svLineEndY[28] = 4275

svLineStartX[29] = 2250                    -- Sensors Fail Message
svLineStartY[29] = 3690
svLineEndX[29] = 2460
svLineEndY[29] = 4230

svLineStartX[30] = 3105                    -- Velocity Vector
svLineStartY[30] = 2280
svLineEndX[30] = 3555
svLineEndY[30] = 1920

svLineStartX[31] = 4080                    -- Airspeed Warning Box
svLineStartY[31] = 1305
svLineEndX[31] = 3645
svLineEndY[31] = 885

svLineStartX[32] = 1230                    -- Vert Spd Scale (VSI)
svLineStartY[32] = 2670
svLineEndX[32] = 540
svLineEndY[32] = 2770

svLineStartX[33] = 1410                    -- VSI Pointer
svLineStartY[33] = 2745
svLineEndX[33] = 1665
```

```
      svLineEndY[33] = 2385

      svLineStartX[34] = 2760                  -- Quad Threat Warning
      svLineStartY[34] = 1635
      svLineEndX[34] = 2310
      svLineEndY[34] = 1785

      svLineStartX[35] = 2535                  -- PROJ / ADJ Indicators
      svLineStartY[35] = 2010
      svLineEndX[35] = 2230
      svLineEndY[35] = 2310

      svLineStartX[36] = 1155                  -- Mode Number
      svLineStartY[36] = 4290
      svLineEndX[36] = 1725
      svLineEndY[36] = 4590


end enterApplication


to handle buttonUp                             -- if user clicks in the
                                                  symbology


      system INT svPagenum                     -- will contain the page
                                                  number of the display
                                                  text viewer
      system INT svPrevPageNum                 -- will contain the
                                                  previous page number
                                                  of the display text
                                                  viewer
      system INT svXlow [35]                   -- coordinate pairs to
                                                  determine if
      system INT svXhigh [35]                  -- button clicked within
                                                  area
      system INT svYlow [35]
      system INT svYHigh [35]
      system svSymbolPageName [35]
      system INT svLineStartX[36]
      system INT svLineStartY[36]
      system INT svLineEndX [36]
      system INT svLineEndY[36]
      local INT temppagenum
      local temppagesuffix

--    initialize the data tables

      get sysmouseposition

      set xcoord to item 1 of it
      set ycoord to item 2 of it

      local matchflag
      local loopindex

      loopindex = 1
```

```
matchflag = 0                    -- if set to 1, indicates successful
                                    button click


      do

         if (((xcoord >= svxlow[loopindex]) and (xcoord <=
         svxhigh[loopindex])) and \
         ((ycoord >= svylow[loopindex]) and (ycoord <=
         svyhigh[loopindex])))
             if loopindex = 35

-- special case for Pitch Ladder/Horizon Line
-- use equation of a line to determine if above or below horizon line
-- equation is based on the coordinates for pitch ladder in svylow,
-- svyhigh, svxlow, svxhigh

                         tempycoord = (-1) * ycoord

-- must negate since coordinate system reversed in y direction

                       if tempycoord > ((810/1245) * xcoord) - 4400

                          tempvar = "4. Pitch Ladder"
                      else
                         if tempycoord < ((810/1245) * xcoord) - 4500
                           tempvar = "4. Pitch Ladder"
                         else
                           tempvar = "5. Horizon Line"
                         end
                      end
                  else
                      tempvar  =  svSymbolPageName[loopindex]
                  end
                  matchflag = 1
                    get pageNumber of page tempvar of book "M2L1Text.Exe"
         else
                   loopindex = loopindex + 1
       end

    until ((loopindex = 36) or (matchflag = 1))


      if matchflag = 1


           svPrevPageNum = svPageNum
           svPageNum = Item 1 of It

       conditions

           when svpagenum < 4
                 tempPageNum = svpagenum
                 tempPageSuffix = " "
           when svpagenum = 4
                 tempPageNum = svpagenum
                 tempPageSuffix = "a"
           when svpagenum = 5
```

```
                tempPageNum = svpagenum-1
                tempPageSuffix = "b"
            when (svpagenum > 5 and svpagenum <25)
                tempPageNum = svpagenum - 1
                tempPageSuffix = " "
            when svpagenum = 25
                tempPageNum = svpagenum - 1
                tempPageSuffix = "a"
            when svpagenum = 26
                tempPageNum = svpagenum - 2
                tempPageSuffix = "b"
            when svpagenum = 27
                tempPageNum = svpagenum - 3
                tempPageSuffix = "c"
            when svpagenum = 28
                tempPageNum = svpagenum - 4
                tempPageSuffix = "d"
            when svpagenum > 28
                tempPageNum = svpagenum - 4
                tempPageSuffix = " "
    end
            if svPrevPageNum > 2
                select line "arrow"
                send clear
                if svPrevPageNum = 5 or svPrevPageNum = 35
                   select line "arrow2"
                   send clear
            end
        end

            sysLineStyle = 1   -- set the style, size, for the line
            sysLineEndStyle = "solidHead,solidtail"
            sysLineEndSize = 1,1
            sysLockScreen = true

            in viewer "Mod2Les1Text"

               go to page svPageNum

        end

        in viewer "Mod2Les1"
            set text of field "Pagenum" to tempPagenum & tempPagesuffix
& " of 33" -- update to current page
                if svPageNum > 2 and svPageNum < 37
                  show group "linepointer"
            else
              hide group "linepointer"
                end
            end

            if svPageNum > 2
             draw line from
svLineStartX[svPageNum],svLineStartY[svPageNum] to
svLineEndX[svPageNum],svLineEndY[svPageNum]
             name of selection = "arrow"
             strokecolor of selection = red
             drawDirect of selection = false
```

```
            if svPageNum = 35
               draw line from 2535,2010 to 2230,2310
               name of selection = "arrow2"
             strokecolor of selection = red
           drawDirect of selection = false
            end
         end

           sysLockScreen = false

       end -- matchflag

end buttonUp

to handle mouseEnter

    system svoldCursor
    system svPageNum
      svoldCursor = sysCursor   -- save in case not within bounds
      get sysmouseposition

      set xcoord to item 1 of it
      set ycoord to item 2 of it

   if ((( xcoord > 1035) and (xcoord <4485)) and ((ycoord > 1035) and
(ycoord < 4485)) and (svPageNum <> 1))

      if object of target is "rectangle" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
    end
      if object of target is "field" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
      end if
      if object of target is "angledline" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
    end
      if object of target is "polygon" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
    end
      if object of target is "irregularpolygon" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
    end
      if object of target is "line" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
    end
      if object of target is "triangle" then
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
      end

      if object of target is "curve" then
         svoldCursor = sysCursor
```

```
        sysCursor = cursor "hand"
    end

    if object of target is "ellipse" then
       if name of target is "Trim"
         svoldCursor = sysCursor
         sysCursor = cursor "hand"
       end
    end

    end
end MouseEnter

to handle mouseLeave
      system svoldCursor
      sysCursor = svoldCursor
end
to handle enterPage

      show viewer "Mod2Les1Text"
      show viewer "Mod2Les1"

end enterPage

to handle leavePage

      hide viewer "M2L1Dpa"

end leavePage
```

INITIAL DISTIBUTION LIST

	Number of Copies
1. Defense Technical Information Center 8725 John J. Kingman Rd., STE 0944 Ft. Belvoir, VA 22060-6218	2
2. Dudley Knox Library Naval Postgraduate School 411 Dyer Rd. Monterey, California 93943-5101	2
3. Kishore Sengupta, Code SM/SE Department of Systems Management Naval Postgraduate School Monterey, California 93943-5002	1
4. Anthony Ciavarelli, Code 10 Aviation Safety Program Naval Postgraduate School Monterey, California 93943-5002	2
5. Commander Naval Air Systems Command (PMA-205) Training Systems 1421 Jefferson Highway (JP-1, Rm.306) Arlington, VA 22243-1205	1
6. LCDR Kenan Shaffer 106 Moran Circle Monterey, CA 93940	1
7. LT Dab Kern 203 Metz Road Seaside, CA 93955	1

11 51NPS
TH 803
1/99 22527-200

DUDLEY KNOX LIBRARY
3 2768 00356570 6